PSYCHOLOGY REVIVALS

Artificial Intelligence

An Introduction

Alan Garnham

REVIVALS

Psychology Revivals

Artificial Intelligence

Originally published in 1988, this book provides a stimulating introduction to artificial intelligence (AI) - the science of thinking machines. After a general introduction to AI, including its history, tools, research methods, and its relation to psychology, Garnham gives an account of AI research in five major areas: knowledge representation, vision, thinking and reasoning, language, and learning. He then describes the more important applications of AI and discusses the broader philosophical issues raised by the possibility of thinking machines. In the final chapter, he speculates about future research in AI, and more generally in cognitive science. Suitable for psychology students, the book also provides useful background reading for courses on vision, thinking and reasoning, language and learning.

Artificial Intelligence

An introduction

Alan Garnham

Routledge
Taylor & Francis Group
LONDON AND NEW YORK

First published in 1988
by Routledge & Kegan Paul

This edition first published in 2018 by Routledge
2 Park Square, Milton Park, Abingdon, Oxon OX14 4RN

and by Routledge
711 Third Avenue, New York, NY 10017

Routledge is an imprint of the Taylor & Francis Group, an informa business

Publisher's Note
The publisher has gone to great lengths to ensure the quality of this reprint but points out that some imperfections in the original copies may be apparent.

Disclaimer
The publisher has made every effort to trace copyright holders and welcomes correspondence from those they have been unable to contact.

A Library of Congress record exists under ISBN: 071020793X

ISBN: 978-1-138-56373-5 (hbk)
ISBN: 978-0-203-70439-4 (ebk)
ISBN: 978-1-138-56914-0 (pbk)

Artificial Intelligence
An introduction

Alan Garnham

Routledge & Kegan Paul
London and New York

First published in 1988 by
Routledge & Kegan Paul Ltd
11 New Fetter Lane, London EC4P 4EE

Published in the USA by
Routledge & Kegan Paul Inc.
in association with Methuen Inc.
29 West 35th Street, New York, NY 10001

Set in Baskerville
by Hope Services Ltd.

Library of Congress Cataloging in Publication Data
Garnham, Alan, 1954–
Artificial intelligence: an introduction / Alan Garnham.
p. cm. — (Introductions to modern psychology)
Bibliography: p.
Includes index.
1. Artificial intelligence. I. Title. II. Series.
Q335.G36 1987
006.3 – dc19

British Library CIP Data also available

ISBN 0–7102–0793–X (c)
0–7102–1416–2 (pbk)

To my mother
in loving memory

Taylor & Francis
Taylor & Francis Group
http://taylorandfrancis.com

Contents

Preface

Artificial Intelligence (AI), the science of thinking machines, is once again a subject of public interest. It has attracted particular attention – and funding – in the United States, but interest has also revived, albeit to a lesser extent, in Britain and the rest of Europe. To make a machine that can 'think', one must have ideas about what thinking is. These ideas provide insights into the way the human mind works. It is, therefore, difficult for psychologists, who have a professional interest in mental phenomena, to ignore AI research. It would, in any case, be wrong for them to do so.

Many psychologists recognise the importance of AI for their own discipline. Its influence is reflected in both research and in teaching. Ideas from AI have had a profound impact on psychological theories, and undergraduate courses in the psychology of vision, thinking and reasoning, and language have included an AI component for some time. Moreover, in recent years the AI content of such courses has increased considerably.

There are several reasons for teaching AI to psychology students in a separate course. I will mention just two. First, AI raises general questions about the explanation of mental phenomena that all psychologists should consider. It is not always appropriate to discuss such questions in specialist courses on, say, vision or psycholinguistics. In any case, it can be difficult to find time. Second, a background in AI, particularly if it is combined with some experience of programming, opens up a range of employment opportunities for psychology graduates. Programming is usually only taught in specialist AI courses. Some departments, among them the Laboratory of Experimental Psychology at Sussex University, have been teaching AI courses for several years. However, they remain in a minority.

There are two approaches to writing a comparatively short textbook. One is to provide a detailed account of a small number of studies, and to assume that a wider range of topics will be discussed in lectures and tutorials. The other is to cover more material in the book, and to allow lecturers and tutors to expand the explanations, where necessary. I have tended to the latter course, so the density of the information in this book is rather high.

When I started writing, my plan – insofar as I had one – was to provide an account of the most important AI programs. As soon as I became conscious of this intention I realised that I did not approve of it. Simply to say 'there is one machine that can do so-and-so and another that can do such-and-such' is of little interest, except to fact collectors. It is not the existence of 'clever' machines that is important to psychologists, but an understanding of how they work and, hence, of the insights into the nature of intelligence that they provide.

I have not, of course, abandoned the idea of an overview of AI. Indeed, since the AI research community is much smaller than that of psychology, and since AI has a shorter history, it is possible for a fairly small book to be relatively comprehensive. Most undergraduate AI courses for computer scientists are taught from a single text, usually called *Artificial Intelligence*. Specialist texts, like specialist courses, remain a rarity. Although this situation is beginning to change, psychology undergraduates, assuming that they will also be referred to the primary literature, are unlikely to need more than a single text.

My misgivings about a simple overview have had three consequences. First, I have tried to emphasise those areas in which AI research has led to general principles for explaining intelligent behaviour in both man and machine. Second, I have given some space to the broader philosophical issues raised by the possibility of intelligent machines, more to stimulate thought than to say the last word on them. Third, I have attempted to give the reader some feel for AI research. For example, chapter 2 contains what some might see as an overelaborate account of the unification and resolution algorithms. In this particular case the intention was to give some idea of the 'nuts and bolts' of AI programs. AI should show that the mind is not inherently mysterious, but if someone cannot see how an idea – in this case the resolution rule for theorem proving – can be incorporated into a program, then the mind will remain a mystery to them.

I have not, however, tried to provided instruction in programming, for two reasons. First, I am not competent to do so. Second, this book is intended for psychologists, and many psychology departments cannot, and will not for some time be able to, teach AI programming. Furthermore, when they are able to, they may not all teach the same language, and it is virtually impossible to provide a programming course that is not language-specific.

The book is part of the series *Introduction to Modern Psychology*. However, I hope that its usefulness will not be restricted to psychologists. It does have a 'psychological' slant, but only insofar as I have emphasised research that is particularly relevant to psychologists. However, I have tried to provide a reasonably fair overview of AI, and one that is accessible to anyone – including 'the educated layman' – who requires an introduction to that discipline. In particular, it may be suitable for those who find books written by computer scientists for computer scientists daunting. The book is intended as a text for the kind of AI course that I expect to become increasingly popular in psychology departments. It could also serve as reading material for AI sections of more traditional courses on vision, thinking and reasoning, language and learning. I hope it will also be of use to psychologists who cannot attend taught courses, but who want a background in AI, be they undergraduates, postgraduates or more senior people.

A number of people have helped make this a better book than I could have produced on my own. Two of them deserve special thanks. The first is Steve Isard, whose knowledge of AI is much broader than mine, and who spent many hours explaining its complexities to me. I often felt that he should

have been the author. He also read through drafts of all the chapters and corrected some of the errors that remained, despite his patient explanations. The second is Jane Oakhill, who also read a complete draft, from the perspective of a potential reader. She helped eliminate many unclarities and stopped me from taking things for granted that I shouldn't have.

Some of the work on chapter 8 was carried out while I was a Visiting Fellow at the Max-Planck-Institüt für Psycholinguistik, at Nijmegen in The Netherlands. I would like to thank the staff there for making my stay a pleasant one and for providing a congenial working environment.

Sussex, January 1987

1 Introduction

Artificial intelligence (AI) is an approach to understanding behaviour based on the assumption that intelligence can best be analysed by trying to reproduce it. In practice, reproduction means simulation by computer. AI is, therefore, part of computer science. Its history is a relatively short one – as an independent field of study it dates back to the mid-1950s. The AI approach contrasts with an older method of studying cognition, that of experimental psychology. Psychology has long had intelligence among its central concerns, intelligence not just as measured in IQ tests, but in the broader sense in which it is required for thinking, reasoning and learning, and in their prerequisites – high-level perceptual skills, the mental representation of information and the ability to use language.

AI and psychology have inevitably interacted with each other. Psychologists have borrowed concepts from AI, and AI workers have taken an interest in psychological findings. Nevertheless, there has been a certain amount of antagonism between the two approaches, with proponents of each pointing out the strengths of their own methodology and the weaknesses of their opponents'. This uneasy relationship lasted until the late 1970s, when many people on both sides felt the need for a more constructive amalgam of these different approaches to the same problems. A new discipline, *cognitive science*, came into being, combining the strengths of psychology, AI and other subjects, in particular linguistics, formal logic and philosophy. Cognitive science attempts to answer some of the unsolved problems about intelligent behaviour, in the widest sense of that term. The importance of its interdisciplinary approach is reflected in the fact that it is now difficult for psychologists to understand new work in perception and cognition if they are ignorant of AI.

This book is an introduction to AI. It describes research in AI, and discusses the strengths and weaknesses of the AI approach to cognition. Only by familiarising themselves with AI can psychologists, and others, judge for themselves what contribution it can make to the study of mental functioning, and in what way it complements and reinforces more traditional psychological techniques.

The present chapter provides a general introduction to AI, its history, its tools and research methods, and its relation to psychology. Chapters 2 to 6 describe AI research in five major areas: knowledge representation, vision, thinking and reasoning, language, and learning. The more important applications of AI are described in chapter 7, and some of its wider implications for a theory of mind are discussed in chapter 8. The final chapter speculates about future research in cognitive science.

What is 'artificial intelligence'?

Artificial intelligence is the study of intelligent behaviour. One of its goals is to understand human intelligence. Another is to produce useful machines. In some ways the term *artificial intelligence* is an unfortunate one. Both parts of it are misleading. On the one hand, as many people have pointed out, *artificial* implies not real. Although many critics of AI have claimed that artificial intelligences are not really intelligent (see chapter 8), most AI researchers disagree with them. On the other hand, the word *intelligence* suggests that AI is restricted to the study of behaviour that is indicative of intelligence, in the everyday sense of that term – behaviour such as solving problems, playing chess and proving theorems in geometry and predicate calculus. These kinds of behaviour are the ones that particularly interested the AI pioneers of the mid-1950s. However, in writing programs to simulate these skills, they developed a battery of programming techniques that could be applied to aspects of behaviour not normally thought of as requiring any great intelligence – recognising objects and understanding simple text, for example. More recently, particularly in the study of visual perception and speech recognition, programming techniques have been introduced that are not immediately applicable to the simulation of problem solving. Nevertheless, it is usual to extend the term *artificial intelligence* to include this work.

Even when AI research aims to reproduce human behaviour, it need not necessarily attempt to reproduce the mechanisms underlying it. AI is more immediately relevant to psychology when it does try to model those mechanisms, but other types of AI research may be of psychological interest if they suggest general principles for describing or modelling cognitive functions. Therefore, although this book is addressed primarily to psychologists, the term *artificial intelligence* will be taken in its most general sense, and the full range of AI research will be discussed.

Artificial intelligence – a brief history

The term *artificial intelligence* was first used in print by John McCarthy, in a proposal for a conference at Dartmouth College, New Hampshire, to discuss the simulation of intelligent behaviour by machines. As an academic discipline, AI had its origins in the mid-1950s, at about the time of the Dartmouth conference. Its history is, therefore, comparatively short. However, AI did not emerge from a theoretical vacuum. Before that time there had been many investigations of the nature of intelligence. However, until the advent of the digital computer, most of this effort was directed towards understanding intelligence as manifested by people.

It is possible, and in many ways useful, to view current AI research as a continuation of previous work in philosophy, science, and technology. Two authors who explore this idea in detail are McCorduck (1979) and Gregory (1981), who lays particular emphasis on the role of technological advances in understanding the human mind.

Perhaps the principal line of development that led to AI was the attempt to produce machines that took the drudgery out of human intellectual endeavour and, at the same time, eliminated some of the errors to which it is prone. Gregory (1981) describes an ancient Greek device, the so-called *Antikythera Mechanism* (c. 80 BC) which models the movements of heavenly bodies. Its discovery shows that the Greeks were more technologically advanced than has often been assumed. However, the gap between ancient Greek culture and our own is comparatively wide. It is difficult to be sure what the existence of such mechanisms meant to the Greeks, or how they shaped ideas about the mind.

The Antikythera Mechanism is the ancestor of medieval orreries and Renaissance clocks. It is not on the direct line to the digital computer. That device, as its name suggests, developed from aids to numerical calculation, which can be traced back through the abacus to groups of pebbles. Calculating *machines*, whose principal components were cogwheels, were first constructed by the philosophers Pascal (1623–1662) and Leibniz (1646–1716). However, the capabilities of these machines were severely limited (by today's standards) by the fact that their parts were mechanical. These same limitations thwarted the ambitions of Charles Babbage (1792–1871) to produce a much more powerful machine. Babbage's first project, the *Difference Engine*, was designed to perform the relatively modest task of compiling tables of logarithms, whose principle use was in nautical computations. In the early nineteenth century these tables were produced by teams of human computers and were often error-ridden, sometimes with fatal results. In the early 1830s, when its construction was almost finished, Babbage lost interest in the Difference Engine, because he had conceived the much more ambitious *Analytical Engine*, which never came close to realisation. The Analytical Engine was intended to compute any mathematical function, not just logarithms. It was to be programmable by punched cards, in much the same way as the recently invented Jacquard loom. As well as performing mathematical computations, Babbage realised that the Analytical Engine would be able to play games such as noughts-and-crosses (tic-tac-toe) and chess, and at one time he proposed to build a game-playing version of the machine to raise funds.

In the event, the all-purpose computer, such as the Analytical Engine was intended to be, did not become a reality until mass-produced electronic components were available. Even then the earliest machines, conceived during the Second World War and constructed shortly afterwards, were unreliable and difficult to operate. The real breakthrough came with the discovery of semi-conductors and the development of the transistor as a replacement for the vacuum-tube diode.

The first programmable computer was constructed in Germany by Konrad Zuse before the war was over (see McCorduck, 1979, p. 50). However, Zuse was not taken seriously by the German authorities, and Germany's defeat meant that his efforts came to nothing. In Britain and the USA, on the other hand, special purpose computing machines had contributed to the war effort and, when the war was over, funding was made available for further development. In the USA, scientists at the Moore School of the University of Pennsylvania had developed the ENIAC, a machine for calculating bombing tables. After the war, they explored its use as a general purpose computer. In Britain, Alan Turing and a team of cryptanalysts at Bletchley Park had used electromagnetic computing machines for code breaking. Only recently has it become public knowledge how crucial this work was in avoiding defeat in the early years of the war. Turing, also, obtained funds for the design of general purpose electronic computers after the war (see Hodges, 1983, for an account of his work).

The first electronic computers, although physically very large, were severely limited in their capabilities compared with even a modest home microcomputer of the late 1980s. Several developments, in addition to the change from valves to transistors, were needed before AI programming became a reality. Two of these were particularly important. The first was the idea of storing a program, rather than just data, in the computer's memory. The second was that of a high-level programming language, from which programs could be translated automatically into a form that the machine could use. This process of translation is called *compilation*.

The first programs that could be called AI programs, though that term had not yet been invented, were game players. Turing in Britain and Shannon (1950) in the USA, like Babbage before them, explored the idea of a chess-playing computer, and Turing had simulated the performance of such a machine by hand before he could program a real computer to play. However, these early projects were hampered by the lack of high-level programming languages. Programming a machine to perform any task was extremely laborious before the invention of such languages.

1956 is a crucial date in the history of AI. That was the year of the conference at Dartmouth College, which its organisers, in particular John McCarthy, hoped would give a large and immediate impetus to AI research. If the effects of the conference were not as dramatic as had been anticipated, with hindsight they still appear highly significant. If nothing else, the use of the term *artificial intelligence* in the proposal for the conference was instrumental in its gaining currency.

Although McCarthy was disappointed that the conference did not produce more immediate results, it brought together many of those who became prominent in the early days of AI, and laid the foundations for work that was very soon underway. Among the people at Dartmouth were Allen Newell and Herbert Simon, who had already implemented a high-level programming language designed for AI research. Using this language they had written a program, called the Logic Theory Machine, which could prove theorems of formal logic (see chapter 4).

In developing this program, Newell and Simon, together with their colleague Shaw, had largely ignored two lines of research that many of the other delegates at the conference considered important. The first was neural nets (McCulloch and Pitts, 1943). Neural nets are models of the logical properties of interconnected sets of nerve cells. By investigating the properties of such nets, neural net theorists hoped to show how the brain could mediate intelligent behaviour. However, this research was based on what later turned out to be a simplistic view of the properties of neurons, and AI researchers soon lost interest in it. Newell, Shaw and Simon emphasised the importance of studying intelligence at a *functional*, rather than a physiological, level, and it is only recently that ideas similar to those of McCulloch and Pitts have been revived in connectionist and parallel distributed processing (PDP) models.

The second idea that Newell, Shaw and Simon repudiated was that programs should work according to the *principles* of formal logic (rather than prove theorems in it). Their claim that people do not reason using logical rules but use *heuristics*, or rules of thumb, remains important to this day.

Although the implications of Newell, Shaw and Simon's work had not become fully apparent by the end of the Dartmouth conference, their ideas became the dominant influence in the early years of AI, as is evidenced by Feigenbaum and Feldman's (1963) *Computers and Thought*, the first collected volume of AI papers, and one which gives a good overview of the early work.

In the early 1960s the Massachusetts Institute of Technology (MIT) became the leading centre for AI research. This period is often referred to as the era of *semantic information processing*, a name taken from a book summarising its most important work (Minsky, 1968). The term *semantic information processing* indicates that the meaning of the information being processed, and not just its structure, is important for the task in hand. For example, in language processing, interest shifted from the syntax-based attempts at machine translation of the 1950s to meaning-based language-understanding systems.

The work of the early 1960s retained the assumption, made by Newell, Shaw and Simon, that AI research should result in general models of intelligent behaviour. Later in the 1960s it was realised that such behaviour requires large amounts of background knowledge, often knowledge that is specific to a particular task. This observation led to the writing of programs that worked in restricted domains, most notably the MIT BLOCKSWORLD, which comprised prismatic blocks on a table top. However, many people continued to believe that principles discovered by solving problems in one domain would carry over to others. This era produced some programs that performed impressively, though it later turned out that they often depended on domain-specific tricks, and that the ideas on which they were based could not, after all, be generalised to other domains.

Another set of programs on which work began in the 1960s were unashamed specialists, whose performance was deliberately restricted to a single type of problem, such as diagnosing a particular class of diseases. These programs, which were intended to be used in the real world, initially met with some hostility – they were not regarded as genuine AI. Now renamed *expert systems*, they form one of the central areas of AI research (see chapter 7).

The 1960s also saw a revival of interest in formal logic as a tool in problem solving. The principal reason for this revival was the invention by Alan Robinson of the resolution method for deriving conclusions from premises stated in predicate calculus (see chapter 4).

The early 1970s was a comparatively quiet period. Although detailed accounts of some of the most impressive semantic information-processing systems were published (e.g. Winograd, 1972) and work on expert systems continued, there was little sense of progress. In Britain, the Lighthill report concluded that AI was not a priority area for research. However, the late 1970s saw a renaissance. After a lengthy period of evolution, the first expert systems were put into everyday use, working out the structure of organic molecules, configuring computer systems and diagnosing diseases. This development signalled that AI had potentially lucrative applications, and was partly responsible for an upsurge in funding. On the theoretical side, much of the work of the preceding twenty-five years was systematised, and a welcome attempt was made to identify underlying principles, and to dispense with *ad hoc* solutions to problems.

Because AI is a young discipline, the age of a piece of research is a poor guide to its current relevance. This book will, therefore, describe some of the earliest work in AI, as well as more recent studies. There will, however, be an emphasis on later work.

The goals of AI research

AI researchers try both to understand intelligent behaviour and to build clever machines. Indeed, a single AI project may have both of these goals. The fact that many projects aim to produce a specific product – a machine that is, or could be, used in the real world – suggests that AI is more like engineering than physics, that it is more of an applied than a pure science (see e.g. Feigenbaum, 1977). The truth is rather that it is more difficult to distinguish between pure and applied AI than between physics and engineering. In some AI projects the primary goal is to produce an intelligent artefact. The principles underlying its behaviour are of secondary importance, and there is no intention to search for new principles. However, in other projects, perhaps the majority, the aim is to understand the mechanisms underlying behaviour, to make general statements about knowledge representation, vision, thinking, language use or learning. In particular, AI programs that try to simulate human behaviour are often written in an attempt to give a principled account of that behaviour.

AI is most akin to engineering when the goal is to produce a program with a specific application, and when there is little or no concern with whether the program solves the problem in the way that a person would. In such cases the existence of a satisfactory working program indicates the successful completion of the project. The distinction between pure and applied AI is blurred by the fact that many AI workers have used this criterion for success even when they were not trying to produce programs with applications in the real world. AI workers write programs that embody general principles, and consider their projects complete when they have a satisfactory program. However, if one's goal is to understand some aspect of language understanding, say, or vision, then the existence of a working program is important only in so far as it illustrates how a set of ideas works in practice. In such a case, the theory of language understanding or vision is embodied in the principles that underlie the construction of the program. Typically these principles will not be sufficient to produce a complete specification for a language-understanding program or a vision program. To make the program work it will be necessary to add components that are of no theoretical interest. It may even be necessary to *patch* the program – to add bits whose sole purpose is to eliminate unforeseen problems and to allow the program to run. Such programming is only useful in so far as it allows a test of whether the rest of the program is consistent and whether it behaves as predicted. It is particularly important that patches should not be allowed to hide problems in underlying principles.

Closely related to the aim of having a complete working program is that of producing an overall model of a cognitive system. One reason for adopting this goal has been the assumption that cognitive systems comprise sets of interactive subsystems, and that this interaction is a major hindrance to testing theories of individual subsystems. If the behaviour of a subsystem depends on that of the rest of the system, then what it does when working alone may give a misleading impression of what it does in the system as a whole. The fallacy in this argument is that even if a subprocessor's operation depends on what is happening elsewhere, it is not necessary to have a complete model of the rest of the system, but only an account of the input/output functions of the other subprocessors. Even if, say, language processing is interactive, a more orthodox scientific approach can be adopted. Tractable subproblems can be identified and solved. Confronting edifices such as the whole of language understanding is not to be recommended.

The goals of having a working program and of modelling a complete cognitive system, when taken together, suggest that an AI program should be a working model of, say, the language-understanding system, and that the program should produce *realistic outputs*. Many AI researchers have deliberately aimed for programs that produce realistic outputs, and they have tried to validate their programs by investigating the range of inputs for which they produce such outputs. This approach can be attacked on two fronts. First, a program that produces realistic outputs may not be a good model of a cognitive system. This problem might be dubbed the ELIZA problem, after Weizenbaum's (1966) program that engages in realistic dialogues with its user (see chapter 5). The program cannot be said to understand what is said to it in any interesting sense. The fact that simulation does not necessarily imply explanation is also reflected in the fact that extremely accurate mechanical clockworks simulating planetary motion were constructed on the basis of the discredited epicyclic theory of that motion. Second, the outputs that a subprocessor should produce can often be specified independently of its role in the system. For example, a parser should output (partial) syntactic analyses. A model of the parser can, therefore, be tested without embedding it in an overall model of the language processor. Hence, realistic outputs are neither necessary nor sufficient for testing hypotheses about the way language is understood.

AI is most directly relevant to psychology when it aims to discover general explanatory principles. Nevertheless, AI applications may shed indirect light on psychological problems, and complete working programs are useful as testbeds for sets of ideas. All these types of research will therefore, be described in this book.

AI and psychology

There are two main reasons why psychologists should study AI. First, and most important, psychology and AI share an interest in the scientific understanding of cognitive abilities such as reasoning, object recognition and language understanding. Psychologists and AI researchers should work together, as cognitive scientists, towards an understanding of human cognition, using whatever techniques are most appropriate. They should not see themselves as being in opposition to each other. Second, in this age of high technology, we make increasing use of intelligent and less intelligent computers in our everyday lives. If our interactions with these machines are to be fruitful, rather than frustrating, system designers must take account of the assumptions people make about how to use computers.

Some AI projects address specifically psychological problems about the way people (or minds, or brains) behave intelligently. Other projects aim to produce clever machines, but do not try to make those machines behave like people. Nevertheless, there is much that psychologists can learn from such research. In many areas of psychology detailed theories have yet to be produced, and any well-specified account of how a complex task can be performed – an account such as might be implicit in a computer program that performs the task – could be a useful source of psychological ideas.

The way people interact with machines, and the assumptions they make about them, are legitimate areas of psychological study. But if psychologists are to contribute to the study of the interaction of people and computers, they have to understand what computers can do. Only on the basis of such understanding can they make sensible suggestions about how computer systems should be designed and how they can be used most effectively. Increasingly, knowing about computers that people have to interact with will mean knowing about AI. So psychologists will need to study AI if they are to contribute to the study of human-computer interaction.

AI as a source of psychological theories

AI had its origins in computer science. Its founders were not inspired by psychological research but by the idea that computers and computer programs might provide an appropriate metaphor for understanding mental processes. However, psychology and AI remained independent of each other for only a short period. In the 1960s the human experimental branch of psychology, at least, was strongly influenced by the computer metaphor for mind, and *information-processing* theories of cognitive functions became the norm. These theories emphasised the role of internal representations and processes in the causation of behaviour, and provided a sharp contrast with previous behaviourist approaches that focused on input and output, and that regarded the 'organism' as a black box whose inner workings were irrelevant to psychological theorising. By the end of the 1960s the term *cognitive psychology* was virtually synonymous with human experimental psychology. This change in approach brought human experimental psychology much closer to AI, whose pioneers had stressed the digital computer's role as a general *information-processing device*, as opposed to a mere number cruncher.

The proliferation of information-processing models in cognitive psychology reflected a very general influence of AI and the computational metaphor on psychological theorising.

The 1960s also saw an attempt to integrate computer modelling much more directly with the experimental study of human behaviour. Newell and Simon (1963) claimed that they could explain the steps that people went through in solving a problem by comparing what they said as they worked on it with how a program, the General Problem Solver (or GPS, see chapter 4), solved the same problem.

Newell and Simon distinguished between the broad field of artificial intelligence and a particular part of it, the computer simulation of human behaviour. AI was the attempt to make machines behave intelligently, by whatever means. *Computer simulation* was a particular method for producing intelligent machines – making them reproduce human behaviour. Computer simulation, therefore, had the specific goal of providing a model of human cognitive functioning. Although the distinction between computer simulation and the rest of AI remains clear in principle, most contemporary AI projects are influenced to a greater or lesser extent by considerations of how people behave. Very few researchers simply attempt to build clever machines. On the other hand, many still have the explicit goal of writing a program that works the way people do. However, the term computer simulation of behaviour is rarely used these days.

To test their computer simulations Newell and Simon introduced a new research methodology – protocol analysis. A protocol is a record, either written or tape-recorded, of what a person is thinking as he or she carries out some experimental task – usually solving a problem. For example, a person working on a geometry proof might say, 'I'm going to drop a perpendicular from point A to line BC, and see if that helps.' Protocols were assumed to be a direct indication of the way people attempted to solve those problems. They were compared with so-called *traces* of a computer program trying to solve the same problem, and an attempt was made to match what the program did with the operations that people said they were performing.

There are two main problems with this technique. First, it is difficult to measure the goodness of fit between a protocol and a trace of a computer program. Second, it is unclear how people's commentary on what they are doing relates to the mental operations that contribute to the solution to a problem. Furthermore, in some cognitive domains, such as object recognition and language understanding, none of the mental operations that underlie our abilities is available to consciousness. In studying these abilities the technique of protocol analysis is inappropriate. However, interest in protocol analysis has recently revived, particularly in so-called naive physics research, which attempts to model the way scientifically unsophisticated people, and scientists in their more relaxed moments, think about the workings of the physical world (see, for example, some of the papers in Gentner and Stevens, 1983).

Work on the computer simulation of behaviour in the 1960s could be regarded as a (failed) attempt by AI researchers to take over cognitive psychology. At the same time psychologists started to take a greater interest in AI research and to import its ideas into their own theories. Despite the slight antagonism between the two disciplines, which arose mainly from a disagreement about how cognitive functions should be studied, the flow of ideas steadily increased. In the late 1960s and early 1970s concepts such as *semantic network*, *production system* and *frame* made their way from AI into psychology. Furthermore, a small but increasing number of psychologists began to express their theories in the form of computer programs (e.g. Norman, Rumelhart and the LNR Research Group, 1975).

A different attempt to link experimental and programming techniques is found in Marr's research on vision (see chapter 3, and Marr, 1982). Marr's work, which in some ways lies outside mainstream AI, shows how psychology and AI might contribute jointly to a science of perceptual and cognitive functioning. One of Marr's most important contributions is his somewhat misleading named concept of a *computational theory* of a psychological process. A computational theory is *not* a theory in the form of a computer program. It is an account at a more abstract level of an information-processing device. It states what function (in the mathematical sense) that device computes, and why it computes that function. Below the computational theory is an algorithmic description of how the function is computed, usually couched in the form of a computer program, and below that details of the hardware on which the program runs. A computational theory is a set of general principles that help to explain a psychological process. Marr, unlike many AI researchers before him, explicitly demands that general principles be formulated. For him a working program does not, by itself, explain psychological functioning.

Cognitive science
By the late 1970s, cognitive psychologists had more in common with AI researchers than with other psychologists, and AI researchers had more in common with cognitive psychologists than with other computer scientists. In recognition of this fact, the term *cognitive science* was coined, and an attempt was made to initiate more interdisciplinary research that combined the strengths of the two approaches.

Interactions between people and artificial intelligences

More and more, computers are becoming part of our everyday lives. Many of these computers, such as those used by utility companies, we do not interact with directly. Others we do, for example those that dispense cash or provide information services over telephone lines. As yet few of these computers show much intelligence. One of the goals of applied AI research is to make them more intelligent so that they are easier to use and so that they can perform more complex tasks than they do at present. For example, a computer that allows access to a large database of information would be simpler to use if it understood questions in ordinary English – if it had an intelligent *natural language front-end.*

To design computers that people can interact with easily it is necessary to know both something about computers and something about human behaviour. Psychologists with a knowledge of AI are, therefore, ideally qualified to provide specifications for such machines. On the one hand, their knowledge of AI indicates to them what such machines are capable of. On the other hand, their psychological background should provide two things; first, a stock of information about typical reactions to machines, and second, a set of techniques for deciding objectively whether a machine is easy to use. It is easy to convince oneself that a machine will be simple to operate, but less sophisticated users may have unanticipated problems.

AI research tools

Controlled experiments play a major role in psychological research on knowledge representation, vision, thinking, language and learning. In these experiments 'stimuli' are presented to human, or sometimes animal, subjects and some aspect of their 'response', either immediate or delayed, to those stimuli is recorded. Such experiments are not conducted in a theoretical vacuum, but are, or at least should be, designed to test predictions derived from theories about the psychological mechanisms underlying our cognitive abilities. Computers play a role in this research, often providing on-line experimental control. However, these computers are comparatively modest – nowadays they are often microcomputers, and their programs only record behaviour. They do not simulate it.

AI research is very different. Part of the difference can be attributed to the applied nature of many AI projects, which aim to produce machines that work in the real world. Nevertheless, pure AI research, as well as applied, contrasts sharply with work in experimental psychology. The most obvious difference is that AI's principal research tool is the digital computer. Furthermore, AI programs are large. Mainframe computers, not micros, are required, so AI research can be very expensive. Such research requires substantial financial backing. In the United States, where most AI research has been carried out, the majority of AI projects have been supported from Defence funds.

Computers

AI is, in principle, possible without digital computers. As Turing demonstrated, the steps that a program goes through can be worked out by hand. However, for all but the simplest of AI programs, such hand simulation is too time-consuming and too error-prone to be useful. In practice, modern AI research, with its complex programs, is impossible without sophisticated hardware. AI workers have always tended to demand the largest available machines, since their programs are both large and take a long time to run. Part of the explanation of this latter fact is the extensive use of recursive functions – functions that in the course of their execution may call themselves over and over again.

Some of the earliest AI programs ran on IBM computers. However, DEC machines have been favoured since the late 1960s, originally DEC-10s (and DEC-20s) and more recently VAXs. It is now also possible to install AI programming languages on home and business microcomputers, and to learn the rudiments of these languages on such machines. However, microcomputers are not suitable for writing and running large-scale programs.

A recent trend is to replace mainframe computers with personal workstations, with a large permanent memory shared between several of them. People with such workstations have, for most purposes, machines dedicated to them, and need not worry about what other users of the system are doing, as they must on a traditional timesharing system. One of the most useful features of a workstation is a large high-resolution graphics screen, on which it is possible to maintain multiple 'windows', so that the programmer can look at several parts of a program at once. This facility is made possible by the powerful processor dedicated to the workstation. A lot of computer power is needed to maintain the complex displays that workstations make available.

Another development is that of special machines on which AI programs run more efficiently, enabling larger projects to be undertaken. *LISP machines* are specially designed to run the programming language LISP. Many of the features of LISP are reflected in the hardware of these machines. There is no need to have programs for converting commands in LISP into statements that the machine understands directly. Most LISP machines are spin-offs from the MIT MACLISP project, which has always emphasised speed and efficiency.

Programming languages

A computer is of little use without programs to run on it, and the writing of programs requires a programming language. Such languages are, therefore, another important tool in AI research.

In one sense all high-level programming languages are equivalent. A mathematical theory – the theory of computability – shows that if a programming language has certain basic features it can perform any computation that can be carried out on any machine (barring limits on the memory of the computer it is running on). However, different high-level programming languages make different types of computation easy, both to think about and to carry out.

Many people regard computers primarily as machines for numerical computations. For these computations languages such as FORTRAN and ALGOL, in scientific applications, and COBOL in business applications, are particularly suitable. However, most of the computations required in modelling intelligent behaviour are *non-numerical* – they require the manipulation of abstract symbols standing not for numbers, but for such things as people, places, words and sentences, and the relations between them. A special class of programming languages has been developed that are particularly suitable for non-numerical computations. They are called *list-processing languages*.

List-processing languages are distinguished from other languages by their use of *list structure*. List structure was first described by Newell, Shaw and Simon (see e.g. Newell and Shaw, 1957), who developed a series of languages called Information-Processing Languages (IPL-I to IPL-V). It is based on the psychological concept of an *association* between two arbitrary objects. A *list* itself is any sequence of elements, some or all of which may themselves be lists. Examples are:

[6 3 7 3 7 4]
[4 5 [5 8 9] [3 4] 3]
[d n e s i f]
[boy girl fish priest]
[2 d 4 r same [x 5] are]
[[the boy] [saw [the girl]]]

The term *list structure* refers to the way lists are represented inside the computer. A *cell* encodes a single association between the first element of a list (called its CAR, in LISP or HEAD, in POP-11) and the rest of it (its CDR or TAIL), which is also a list. Another cell encodes the association between that list's HEAD and TAIL. The list [2 4 6] is, therefore, represented as three cells. The first associates 2 with the list [4 6]. The second associates 4 with [6] (the list containing the single element 6). The final cell associates the element 6 with [] (the empty list).

In the 1960s the IPLs were replaced by LISP (LISP-Processing language), which had been invented by McCarthy in 1958, and which rapidly became the most important AI language, at least in the USA.

LISP has a number of features that make it particularly suitable for AI programming. First, like the versions of BASIC that run on home micros, LISP is primarily an *interpretive language*. There is no need to go through an intermediate process of compilation between typing the text of the program and getting it to run. LISP programs can be compiled, and they usually are once they have been written and debugged, but only because compiled programs run faster than interpreted ones. Second, because LISP is interpretive it allows interactive programming. A programmer can type a statement that will be interpreted straight away, or define a procedure and try that out immediately. Third, a LISP program comprises a large number of separate procedures, each of which can be written and debugged on its own. This feature further enhances the value of interactive programming. Fourth, defining recursive procedures – procedures that call themselves – is straightforward in LISP. Recursive procedures play a crucial role in AI programs. A very simple recursive procedure is one for calculating factorial n (i.e. n × n -1 × n-2 × . . . × 3 × 2 × 1). Informally this procedure is:

```
to find factorial n
  if n = 1 then the answer is 1
  otherwise to get the answer
                multiply the factorial of n−1 by n.
```

In computing, say, factorial 4, this procedure will call itself to calculate factorial 3, factorial 2 and factorial 1. Only when it calculates factorial 1 will it get a definite result (1), which is then fed back into the other calculations, whose results are still pending.

Another distinctive feature of LISP is that LISP programs are themselves lists, so the syntax of LISP is simple. However, this simplicity does not mean that LISP programs are readily comprehensible. Indeed, one frequent complaint is that they are hard to understand. Another is that there has never been a standard for LISP in the way that there has been for languages such as FORTRAN and ALGOL, so it may be difficult to transfer a LISP program from one machine to another. However, two dialects of LISP are more important than any others – MACLISP, developed at MIT, and INTERLISP, developed at BBN (Bolt, Berenek & Newman) and Xerox. Each of these is far more than a language, it is also a programming environment. The differences between the two are not so much in the language itself, though that does differ, but in the additional facilities provided. A recent attempt to provide a subset of LISP in which programs can be written that run on any LISP system has resulted in the language COMMON LISP.

LISP interpreters (and compilers) are usually written in LISP, so a LISP statement or procedure will typically be interpreted by another LISP procedure. LISP, therefore, lends itself to having additional facilities designed to run 'on top of' it, because a LISP programmer can write a procedure to interpret a new command. These additional facilities sometimes become so complex that they are best thought of as new languages. The most important are PLANNER and CONNIVER (see chapter 4).

In Britain the language POP-2 (Burstall, Collins and Popplestone, 1971) was developed as an alternative to LISP. In later versions it is known as POP-11 (Barrett, Ramsay and Sloman, 1985). POP has all of the advantages of LISP, except that it sacrifices simplicity of syntax for readability. Its syntax is modelled on that of ALGOL, and POP programs are not lists. POP programs are easier to read than equivalent LISP programs. In many cases they are quite close to English. The dialect of POP-11 that runs in the POPLOG programming environment is likely to become a standard.

Another language that is now popular for some AI applications is PROLOG (Clocksin and Mellish, 1981). Considerable attention has focused on this language since the announcement that the Japanese intended to write intelligent knowledge-based systems in PROLOG as part of their fifth-generation project. PROLOG is based on a subset of the logical language predicate calculus, and its use is sometimes referred to as *logic programming*. In a logic programming language, a program is a set of logical formulae and the result of running the program is a deductive consequence of those formulae. A logic programming language, therefore, requires a built-in theorem prover to deduce the consequences of its programs. PROLOG provides only an approximation to true logic programming, because the behaviour of a PROLOG program cannot be completely predicted from the corresponding set of logical formulae. Part of the reason is that PROLOG's theorem prover is comparatively simple. While it is adequate for some applications, it has to be programmed around in others.

Programming environments

Present-day computer programmers, particularly those who produce large programs, are not satisfied with a language for writing their programs and a computer to run them on. They want a range of other facilities for editing and debugging programs, and for recompiling the relevant parts of a program when small changes are made. Since AI programs have always been among the largest and most complex that have been written, the demand for such advanced facilities has come mainly from the AI community. They are provided in the MACLISP, INTERLISP and POPLOG *programming environments* already mentioned. The range of such facilities has increased considerably since the early days of AI research, and modern programming environments remove many of the chores from computer programming. AI research has not suddenly become easy, but the amount of time spent on menial tasks has been dramatically reduced. A good programming environment is now one of the standard tools of AI research.

One of the most important facilities in an AI programming environment is an editor specially suited to LISP or POP-11. The best known of these is the EMACS editor developed at MIT for MACLISP. The POPLOG POP editor is called VED. Such an editor knows the names of the built-in functions of the language and its syntax. It can, therefore, detect possible spelling and syntax errors. Other facilities include on-line documentation and help in keeping track of all the procedures in a large program.

Summary

AI represents a comparatively recent approach to the study of cognition. It is part of computer science, and aims to simulate intelligent behaviour, in a very broad sense of that term, using computer programs. The approach of AI can be contrasted with that of psychology, which proceeds via the experimental testing of predictions derived from theories.

AI has many intellectual antecedents, in particular the work of Babbage, but real progress was impossible before the advent of the digital computer. Such machines were developed in the years after the Second World War, and by the mid-1950s AI had become an actuality with Newell, Shaw and Simon's Logic Theory Machine. Newell *et al.* emphasised functional, as opposed to physiological, models in AI, and introduced the concept of heuristics, or rules of thumb, for solving problems. In the 1960s the range of AI projects increased, the idea of semantic information processing was introduced, and work began on expert systems. More recently, attempts have been made to combine the advantages of AI and psychology in cognitive science.

Pure and applied AI research have not always been easy to disentangle, partly because many AI projects adopt the goal of a working program with realistic outputs. However, in the absence of underlying principles, such programs may make little contribution to the understanding of intelligent behaviour. A recent trend in AI is to place more emphasis on such principles.

Psychologists should study AI for two reasons. First, and most important, AI is a source of psychological theories. Concepts from AI have regularly been imported into psychology, but AI also shows how precise and detailed models of cognitive functioning can be formulated. Second, psychologists who know about AI are uniquely placed to make decisions about the deployment of information technology, because they know both what computers are capable of and how people will react to them.

The mainstay of AI research is the writing of computer programs. Its principal research tools are, therefore, large computers, special list-processing computer languages – primarily LISP, POP-11 and PROLOG – and programming environments designed to make the routine parts of programming easier and less time-consuming.

References

Allen, J.F. and Perrault, C.R. (1980), 'Analyzing intentions in dialogues', Artificial Intelligence, vol. 15, pp. 143178.

Angluin, D. (1978), 'Inductive inference of formal languages from positive data', Information and Control, vol. 45, pp. 117-135.

Appelt, D.E. (1985), 'Planning English referring expressions', Artificial Intelligence, vol. 26, pp. 1-33.

Ballard, D.H. (1984), 'Parameter nets', Artificial Intelligence, vol. 22, pp. 235-267.

Barr, A. and Feigenbaum, E.A. (1981), The Handbook of Artificial Intelligence, vol. 1, Los Altos, Calif., William Kaufmann.

Barr, A. and Feigenbaum, E.A. (1982), The Handbook of Artificial Intelligence, vol. 2, Los Altos, Calif., William Kaufmann.

Barrett, R. . Ramsay, A. and Sloman, A. (1985), POP-ll: A Practical Language for Artificial Intelligence, Chichester, Ellis Horwood.

Barrow, H.G. and Tenenbaum, J.M. (1978), 'Recovering intrinsic scene characteristics from images', in A. R. Hanson and . M. Reisman (eds), Computer Vision Systems, New York, Academic Press.

Bartlett, F.C. (1932), Remembering: A Study in Experimental and Social Psychology, Cambridge, Cambridge University Press.

Berliner, H. and Campbell, M. (1984), 'Using chunking to solve chess pawn endgames', Artificial Intelligence, vol. 23, pp. 97120,

Bernstein, . , Arbuckle, T. , Roberts, M. de V. and Belsky, M.A. (1958), 'A chess-playing program for the IBM 704 computer', Proceedings of the Western Joint Computer Conference, vol. 16, pp. 157-159.

Berwick, R.C. (1983), 'Computational aspects of discourse', in M. Brady and R. C. Berwick (eds), Computational Models of Discourse, Cambridge, Mass., MIT Press.

Berwick, R.C. (1985), The Acquisition of Syntactic Knowledge, Cambridge, Mass., MIT Press.

250 Bledsoe, W.W. (1977), 'Non-resolution problem solving', Artificial intelligence, vol. 9, pp. 1-35.

Bobrow, D.G. (1968), 'Natural language input for a computer problem-solving system', in M. Minsky (ed.), Semantic Information Processing, Cambridge, Mass., MIT Press.

Bobrow, D.G. and Collins, A.M. (eds) (1975), Representation and Understanding, New York, Academic Press.

Bobrow, D.G. and Winograd, T. (1977), 'An overview of KRL: a knowledge representation language', Cognitive Science, vol. 1, pp. 3-46.

Bobrow, D.G. , Kaplan, R.M. , Kay, M. , Norman, D.A. , Thompson, H. and Winograd, T. (1977), 'GUS, a frame-driven dialog system', Artificial Intelligence, vol. 8, pp. 155-173.

Bowden, M. (1977), Artificial Intelligence and Natural Man, Hassocks, Harvester.

Bower, G.H. , Black, J.B. and Turner, T.J. (1979), 'Scripts in memory for text', Cognitive Psychology, vol. 11, pp. 177-220.

Boyer, R.S. and Moore, J.S. (1979), A Computational Logic, New York, Academic Press.

Brachman, R.J. (1979), 'On the epistemologica! status of semantic networks', in . V. Findler (ed.), Associative Networks: Representation and Use of Knowledge by Computers, New York, Academic Press.

Brady, J.M. (1984), 'Intelligent robots: connecting perception to action', in P. H. Winston and K. A. Prendergast (eds), The AI Business: The Commercial Uses of Artificial Intelligence, Cambridge, Mass., MIT Press.

Bratko, I. , Kopec, D. and Michie, D. (1978), 'Pattern-based representation of chess end-game knowledge', Computer Journal, vol. 21, pp. 149-153.

Bridle, J.S. , Brown, M.D. and Chamberlin, R.M. (1982), one-pass algorithm for connected word recognition', Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 899902.

Broadbent, D. (1985), question of levels: comment on McClelland and Rumelhart', Journal of Experimental Psychology: General, vol. 114, pp. 189-192.
Brown, J.S. , Burton, R.R. and de Kleer, J. (1982), 'Pedagogical, natural language and knowledge engineering techniques in SOPHIE I, II, and III, in S. Derek and J.S. Brown (eds), Intelligent Tutoring Systems, New York, Academic Press.
Bruce, B.C. (1982), 'Natural communication between person and computer', in W.G. Lehnert and M.H. Ringle (eds), Strategies for Natural Language Processing, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Bundy, A. (ed.) (1978), Artificial Intelligence: An Introductory Course, Edinburgh, Edingburgh University Press.
Bundy, A. (1983), The Computer Modelling of Mathematical Reasoning, London, Academic Press.
251 Bundy, A. and Welham, . (1981), 'Using meta-level inference for selective application of multiple rewrite rules in algebraic multiplication', Artificial Intelligence, vol. 16, pp. 189-211.
Burstall, R.M. , Collins, J.S. and Popplestone, R.J. (1971), Programming in POP-2, Edinburgh, Edinburgh University Press.
Carbonell, J.G. (1982), 'Metaphor: an inescapable phenomenon in natural-language comprehension', in W. G. Lehnert and M. H. Ringle (eds), Strategies for Natural Language Processing, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Carbonell, J.R. (1970), 'AI in CAI: an artificial intelligence approach to computer-aided instruction', IEEE Transactions on Man-Machine Systems, vol. 11, pp. 190-202.
Charniak, E. and McDermott, D.V. (1985), Introduction to Artificial Intelligence, Reading, Mass., Addison-Wesley.
Charniak, E. , Riesbeck, C.K. and McDermott, D.V. (1980), Artificial Intelligence Programming, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Chomsky, N. (1957), Syntactic Structures, The Hague, Mouton.
Chomsky, N. (1959), review of B. F. Skinner's "Verbal Behavior"', Language, vol. 35, pp. 26-58.
Chomsky, N. (1965), Aspects of the Theory of Syntax, Cambridge, Mass., MIT Press.
Chomsky, N. (1977), 'On wh-movement', in P. W. Culicover , T. Wasow and A. Akmajian (eds), Formal Syntax, New York, Academic Press.
Chomsky, N. (1981), Lectures on Government and Binding, Dordrecht, Foris.
Chomsky, N. and Lasnik, H. (1977), 'Filters and control', Linguistic Inquiry, vol. 8, pp. 425-504.
Clancey, W.J. (1984), 'Methodology for building an intelligent tutoring system', in W. Kintsch , J. R. Miller and P. G. Poison (eds), Methods and Tactics in Cognitive Science, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Clocksin, W. and Mellish, C. (1981), Programming in Prolog, Berlin, Springer-Verlag.
Clowes, M.B. (1971), 'On seeing things', Artificial Intelligence, vol. 2, pp. 79-116.
Cohen, Paul R. and Feigenbaum, E.A. (1982), The Handbook of Artificial Intelligence, vol. 3, Los Altos, Calif., William Kaufmann.
Cohen, Philip R. and Perrault, C.R. (1979), 'Elements of a plan-based theory of speech acts', Cognitive Science, vol. 3, pp. 177-212.
Cohen, Philip R. , Perrault, C.R. and Allen, J.F. (1982), 'Beyond question answering', in W. G. Lehnert and M. H. Ringle (eds), Strategies for Natural Language Processing, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Collins, A.M. (1976), 'Processes in acquiring knowledge', in R. C. Anderson , R. J. Spiro and W. E. Montague (eds), Schooling and the Acquisition of Knowledge, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
252 Collins, A.M. and Loftus, E.F. (1975), ' spreading-activation theory of semantic processing', Psychological Review, vol. 82, pp. 407-428.
Collins, A.M. and Quillian, M.R. (1969), 'Retrieval time from semantic memory', Journal of Verbal Learning and Verbal Behavior, vol. 8, pp. 240-247.

Collins, A.M. and Quillian, M.R. (1972), 'Experiments on semantic memory and language comprehension', in L. W. Gregg (ed.), Cognition in Learning and Memory, New York, Wiley.
Collins, N.L. and Michie, D. (1967), Machine Intelligence 1, Edinburgh, Edinburgh University Press.
Grain, S. and Steedman, M.J. (1985), 'On not being led up the garden path: the use of context by the psychological parser', in D. Dowty , L. Karttunen and A. Zwicky (eds), Natural Language Parsing, Cambridge, Cambridge University Press.
Cullingford, R.E. (1978), Script Application: Computer Understanding of Newspaper Stories, Yale University Department of Computer Science, Report no. 116.
Davis, R. (1982), 'TEIRESIAS: applications of meta-level knowledge', in R. Davis and D.B. Lenat (eds), Knowledge-Based Systems in Artificial Intelligence, New York, McGraw-Hill.
Davis, R. (1984), 'Amplifying expertise with expert systems', in P. H. Winston and K. A. Prendergast (eds), The AI Business: The Commercial Uses of Artificial Intelligence, Cambridge, Mass., MIT Press.
Davis, R. , Austin, H. , Carlbom, I. , Frawley, B. , Pruchnik, P. , Sneiderman, R. and Gilreath, A. (1981), 'The DIPMETER ADVISOR: interpretation of geological signals', Proceedings of the Seventh International Joint Conference on Artificial Intelligence, pp. 846-849.
Dietterich, T.G. and Michalski, R. (1981), 'Inductive learning of structural descriptions: evaluation criteria and comparative review of selected methods', Artificial Intelligence, vol. 16, pp. 257-294.
Doyle, J. (1979), truth-maintenance system', Artificial Intelligence, vol. 12, pp. 231-272.
Draper, S.W. (1981), 'The use of gradient and dual space in line-drawing interpretation', Artijkial Intelligence, vol. 17, pp. 461-508.
Dreyfus, H.L. (1979), What Computers Can't Do, 2nd edn, New York, Harper & Row.
Duda, R.O. , Hart, P.E. , Nilsson, N.J. and Sutherland, G.L. (1978), 'Semantic network representations in rule based inference systems', in D. A. Waterman and F. Hayes-Roth (eds), Pattern Directed Inference Systems, New York, Academic Press.
Dyer, M.G. (1983), In-Depth Understanding: A Computer Model of Integrated Processing for Narrative Comprehension, Cambridge, Mass., MIT Press.
253 Earley, J. (1970), 'An efficient context-free parsing algorithm', Communications of the Association for Computing Machinery, vol. 13, pp. 94-102.
Erman, L.D. and Lesser, V.R. (1980), 'The HEARSAY-II speech understanding system', in W. A. Lea (ed.), Trends in Speech Recognition, Englewood Cliffs, New Jersey, Prentice-Hall.
Ernst, G. and Newell, A. (1969), GPS: A Case Study in Generality and Problem Solving, New York, Academic Press.
Etherington, D.W. (1987), 'Formalizing nonmonotonic reasoning systems', Artificial Intelligence, vol. 31, pp. 41-85,
Evans, T.G. (1968), ' program for the solution of geometric-analogy intelligence test questions', in M. Minsky (ed.), Semantic Information Processing, Cambridge, Mass., MIT Press.
Fahlman, S.E. (1979), NETL: A System for Representing and Using Real-World Knowledge, Cambridge, Mass., MIT Press.
Falk, G. (1972), 'Interpretation of imperfect line data as a three-dimensional scene', Artificial Intelligence, vol. 3, pp. 101-144.
Feigenbaum, E.A. (1977), 'The art of artificial intelligence, 1: Theories and case studies in knowledge engineering', Proceedings of the Fifth International Joint Conference on Artificial Intelligence, pp. 10141029.
Feigenbaum, E.A. and Feldman, J. (eds) (1963), Computers and Thought, New York, McGraw-Hill.

Feldman, J.A. and Ballard, D.H. (1982), 'Connectionist models and their properties', Cognitive Science, vol. 6, pp. 205-254.
Fikes, R.E. and Nilsson, N.J. (1971), 'STRIPS: a new approach to the application of theorem proving to problem solving', Artificial Intelligence, vol. 2, pp. 189-208.
Fillmore, C.J. (1968), 'The case for case', in E. Bach and R. T. Harms (eds), Universals in Linguistic Theory, New York, Holt, Rinehart & Winston.
Fillmore, C.J. (1971), 'Types of lexical information', in D. D. Steinberg and L. A. Jakobovits (eds), Semantics: An Interdisciplinary Reader in Philosophy, Linguistics and Psychology, Cambridge, Cambridge University Press.
Findler, N.V. (ed.) (1979), Associative Networks: Representation and Use of Knowledge by Computers, New York, Academic Press.
Fodor, J.A. (1980), 'Methodological solipsism considered as a research strategy in cognitive psychology', Behavioral and Brain Sciences, vol. 3, pp. 63-73.
Frazier, L. and Fodor, J.D. (1978), 'The sausage machine: a new two-stage parsing model', Cognition, vol. 6, pp. 291-325.
Frege, G. (1879/1972), 'Conceptual notation: a formula language of pure thought modelled upon the formula language of arithmetic', in T.W. Bynum (ed. and trans.), Conceptual Notation and Related Articles, Oxford, Oxford University Press. (First published in German in 1879, Halle, L. Nerbert.)
Frisby, J.P. and Mayhew, J.E.W. (1980), 'Spatial frequency and tuned channels: implications for structure and function from psychophysical and computational studies of stereopsis', Philosophical Transactions of the Royal Society of London, Series B, vol. 290, pp. 95-116.
254 Garnham, A. (1985), Psycholinguistics: Central Topics, London, Methuen.
Gazdar, G. , Klein, E. , Pullum, G. and Sag, I. (1985), Generalized Phrase Structure Grammar, Oxford, Blackwell.
Gelernter, H.L. (1963), 'Realisation of a geometry-theorem proving machine', in E.A. Feigenbaum and J. Feldman (eds), Computers and Thought, New York, McGraw-Hill.
Gelernter, H.L. , Sanders, A.F. , Larsen, D.L. , Agarival, K.K. , Boivie, R.H. , Spritzer, G.A. and Searleman, J.E. (1977), 'Empirical explorations of SYNCHEM', Science, vol. 197, pp. 1041-1049.
Gentner, D. and Stevens, A.L. (1983), Mental Models, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Gilmore, P.C. (1980), 'An examination of the geometry theorem machine', Artificial Intelligence, vol. 2, pp. 171-187.
Gold, E.M. (1967), 'Language identification in the limit', Information and Control, vol. 10, pp. 447474.
Goldstein, I.P. (1975), ' summary of MYCROFT; a system for understanding simple picture programs', Artificial Intelligence, vol. 6, pp. 249-288.
Graesser, A.C. , Gordon, S.E. and Sawyer, J.D. (1979), 'Recognition memory for typical and atypical actions in scripted activities: tests of a script pointer + tag hypothesis', Journal of Verbal Learning and Verbal Behavior, vol. 18, pp. 319-332.
Grape, G.R. (1969), Computer Vision Through Sequential Abstractions, Stanford, Calif., Stanford University Department of Artificial Intelligence.
Green, B.F. , Wolf, A.K. , Chomsky, C. and Laughery, K. (1961), 'BASEBALL: an automatic question answerer', Proceedings of the Western Joint Computer Conference, vol. 19, pp. 219-224.
Green, C. (1969), 'Application of theorem proving to problem solving', Proceedings of the First International Joint Conference on Artificial Intelligence, pp. 219-239.
Green, C. (1977), ' summary of the PSI program synthesis system', Proceedings of the Fifth International Joint Conference on Artificial Intelligence, pp. 380 381.
Green, C. and Westfold, S. (1982), 'Knowledge-based programming self applied', in J. E. Hayes , D. Michie and Pao Y-. (eds), Machine Intelligence 10, Chichester, Ellis Horwood.
Greenblatt, R.D. , Eastlake, D.E. and Crocker, S.D. (1967), 'The Greenblatt chess program', Proceedings of the American Federation of Information Processing Societies Fall Joint Computer Conference, vol. 31, pp. 801-810.

Gregory, R.L. (1981), Mind in Science: A History of Explanations in Psychology and Physics, London, Weidenfeld & Nicolson.
Grice, H.P. (1975), 'Logic and conversation', in P. Cole and J. L. Morgan (eds), Syntax and Semantics, 3: Speech Acts, New York, Seminar Press.
255 Grimes, J. (1975), The Thread of Discourse, The Hague, Mouton.
Grosz, . (1981), 'Focusing and description in natural language dialogues', in A. K. Joshi , B. L. Webber and I. A. Sag (eds), Elements of Discourse Understanding, Cambridge, Cambridge University Press.
Guzman, A. (1968), 'Decomposition of a visual scene into three-dimensional bodies', Proceedings of the American Federation of Information Processing Societies Fall Joint Computer Conference, vol. 33, pp. 291304.
Haas, A.R. (1986), ' syntactic theory of belief and action', Artificial Intelligence, vol. 28, pp. 245292.
Halliday, M.A.K. (1970), 'Language structure and language function', in j. Lyons (ed.), New Horizons in Linguistics, Harmondsworth, Penguin.
Hart, P.E. , Nilsson, N.J. and Raphael, B. (1968), ' formal basis for the heuristic determination of minimum cost paths', IEEE Transactions on Systems Science and Cybernetics, vol. 4, pp. 100-107.
Haugeland, J. (1985), Artificial Intelligence: The Very Idea, Cambridge, Mass., MIT Press.
Heldhorn, G.E. (1975), Simulation Programming through Natural Language Dialogue, Amsterdam, North-Holland.
Hendrix, G.G. (1979), 'Encoding knowledge in partitioned networks', in N.V. Findler (ed.), Associative Networks: Representation and Use of Knowledge by Computers, New York, Academic Press.
Hewitt, C. (1972), Description and Theoretical Analysis (Using Schemata) of PLANNER: A Language for Proving Theorems and Manipulating Models in a Robot, Massachusetts Institute of Technology, Laboratory of Artificial Intelligence, Technical Report AI-TR-258.
Hillis, W.D. (1985), The Connection Machine, Cambridge, Mass., MIT Press.
Hinton, G.E. (1981), 'Implementing semantic networks in parallel hardware', in G. E. Hinton and J. A. Anderson (eds), Parallel Models of Associative Memory, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Hodges, A. (1983), Alan Turing: The Enigma of Intelligence, London, Burnett.
Holmes, J.N. , Mattingly, I.G. and Shearme, J.N. (1964), 'Speech synthesis by rule', Language and Speech, vol. 7, pp. 127-143.
Horn, B.K.P. (1973), The Binford-Horn Edge Finder, Massachusetts Institute of Technology, Laboratory of Artificial Intelligence, Memo no. 285.
Horn, B.K.P. (1975), 'Obtaining shape from shading information', in P. H. Winston (ed.), The Psychology of Computer Vision, New York, McGraw-Hill.
Horn, B.K.P. (1977), 'Understanding image intensities', Artificial Intelligence, vol. 8, pp. 201-231.
Houghton, G. and Isard, S.D. (in press), 'Why to speak, what to say, and how to say it', in P. Morris (ed.), Modelling Cognition, Chichester, Wiley.
Huffman, D.A. (1971), 'Impossible objects as nonsense sentences', in B. Meitzer and D. Michie (eds), Machine Intelligence 6, Edinburgh, Edinburgh University Press.
256 Hunt, E.B. (1975), Artificial Intelligence, New York, Academic Press.
Ikeuchi, K. (1984), 'Shape from regular patterns', Artificial Intelligence, vol. 22, pp. 49-75.
Ikeuchi, K. and Horn, B.K.P. (1981), 'Numerical shape from shading and occluding boundaries', Artificial Intelligence, vol. 17, pp. 141-184.
Isard, S.D. (1986), 'Levels of representation in computer speech synthesis and recognition', in M. Yazdani (ed.), Artificial Intelligence: Principles and Applications, London, Chapman & Hall.
Isard, S.D. and Miller, D.A. (1986), 'Diphone synthesis techniques', Proceedings of the IEEE International Conference of Speech Input/Output Techniques and Applications, pp. 77-82.

Johnson-Laird, P.N. (1977), 'Procedural semantics', Cognition, vol. 5, pp. 189-214.
Johnson-Laird, P.N. (1983), Mental Models: Towards a Cognitive Science of Language, Inference and Consciousness, Cambridge, Cambridge University Press.
Johnson-Laird, P.N. and Garnham, A. (1980), 'Descriptions and discourse models', Linguistics and Philosophy, vol. 3, pp. 371-393.
de Jong, G. (1982), 'An overview of the FRUMP system', in W. G. Lehnert and M. H. Ringle (eds), Strategies for Natural Language Processing, Hillsdale, New Jersey, Lawrence Erlbaum Associates,
Joshi, A.K. (1985), 'Tree adjoining grammars: how much context-sensitivity is required to provide reasonable structural descriptions?', in D. R. Dowty , L. Karttunen and A. M. Zwicky (eds), Natural Language Parsing, Cambridge, Cambridge University Press.
Julesz, B. (1971), Foundations of Cyclopean Perception, Chicago, University of Chicago Press.
Kanade, T. (1980), ' theory of the Origami world', Artificial Intelligence, vol. 13, pp. 279-311.
Kaplan, R.M. (1973), ' general syntactic processor', in R. Rustin (ed.), Natural Language Processing, Englewood Cliffs, New Jersey, Prentice-Hall.
Kaplan, R.M. and Bresnan, J. (1982), 'Lexical-functional grammar: a formal system for grammatical representation', in j. Bresnan (ed.), The Mental Representation of Grammatical Relations, Cambridge, Mass., MIT Press.
Kay, M. (1973), 'The MIND system', in R. Rustin (ed.), Natural Language Processing, Englewood Cliffs, New Jersey, Prentice-Hall.
Kimball, J. (1973), 'Seven principles of surface structure parsing in natural language', Cognition, vol. 2, pp. 15-47.
Kister, J. , Stein, P. , Ulam, S. , Waiden, W. and Wells, M. (1957), 'Experiments in chess', Journal of the Association for Computing Machinery, vol. 4, pp. 174177.
Korf, R.K. (1985), 'Macro-operators: a weak method for learning', Artificial Intelligence, vol. 26, pp. 35-77.
257 Kowalski, R. (1979), Logic for Problem Solving, Amsterdam, North-Holland.
Kraft, A. (1984), 'XCON: an expert configuration system', in P. H. Winston and K. A. Prendergast (eds), The AI Business: The Commercial Uses of Artificial Intelligence, Cambridge, Mass., MIT Press.
Land, E.H. and McCann, J.J. (1971), 'Lightness and retinex theory ', Journal of the Optical Society of America, vol. 61, pp. 111.
Lehnert, W. (1982), 'Plot units: a narrative summarization structure', in W. G. Lehnert and M. H. Ringle (eds), Strategies for Natural Language Processing, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Lenat, D.B. (1977), 'The ubiquity of discovery', Artificial Intelligence, vol. 9, pp. 257-285.
Lenat, D.B. (1982), 'AM: discovery in mathematics as heuristic search', in R. Davis and D. B. Lenat (eds), Knowledge-Based Systems in Artificial Intelligence, New York, McGraw-Hill.
Lenat, D.B. (1983), 'EURISKO: a program that learns new heuristics and domain concepts', Artificial Intelligence, vol. 21, pp. 61-98.
Lenat, D.B. and Brown, J.S. (1984), 'Why AM and EURISKO appear to work', Artificial Intelligence, vol. 23, pp. 269-294.
Lettvin, J.Y. , Maturana, H.R. , McCulloch, W.S. and Pitts, W.H. (1959), 'What the frog's eye tells the frog's brain', Proceedings of the Institute of Radio Engineers, vol. 47, pp. 1940-1951.
Levesque, H. and Mylopoulos, J. (1979), ' procedural semantics for semantic networks', in . V. Findler (ed.), Associative Networks: Representation and Use of Knowledge by Computers, New York, Academic Press.
Lindsay, R. , Buchanan, B.G. , Feigenbaum, E.A. and Lederberg, J. (1980), Applications of Artificial Intelligence for Chemical Inference: The DENDRAL Project, New York, McGraw-Hill.

Linggard, R. (1985), Electronic Synthesis of Speech, Cambridge, Cambridge University Press.
Lowerre, B. and Reddy, D.R. (1980), 'The HARPY speech understanding system', in W. A. Lea (ed.), Trends in Speech Recognition, Englewood Cliffs, New Jersey, Prentice-Hall.
McCarthy, J. (1980), 'Circumspection - a form of non-monotonic reasoning', Artificial Intelligence, vol. 13, pp. 27-39.
McCarthy, J. (1986), 'Applications of circumspection to formalizing common-sense knowledge', Artificial Intelligence, vol. 28, pp. 89-116.
McClelland, J.L. and Rumelhart, D.E. (1981), 'An interactive activation model of context effects in letter perception, Part 1: An account of basic findings', Psychological Review, vol. 88, pp. 375-407.
McClelland, J.L. and Rumelhart, D.E. (1985), 'Distributed memory and the representation of general and specific information', Journal of Experimental Psychology: General, vol. 114, pp. 159-188.
258 McClelland, J.L. and Rumelhart, D.E. (1986), Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol. 2: Psychological and Biological Models, Cambridge, Mass., MIT Press.
McCorduck, P. (1979), Machines Who Think: A Personal Inquiry into the History and Prospects of Artificial Intelligence, San Francisco, Freeman.
McCulloch, W.S. and Pitts, W.H. (1943), ' logical calculus of ideas immanent in nervous activity', Bulletin of Mathematical Biophysics, vol. 5, pp. 115-133.
McDermott, D.V. and Doyle, J. (1980), 'Non-monotonic logic I', Artificial Intelligence, vol. 13, pp. 41-72.
McDermott, D.V. and Sussman, G.J. (1972), The CONNIVER Reference Manual, Massachusetts Institute of Technology, Laboratory of Artificial Intelligence, Memo no. 259.
McDermott, J. (1982), 'RI: A rule-based configurer of computer systems', Artificial Intelligence, vol. 19, pp. 39-88.
McKeown, K.R. (1985), 'Discourse strategies for generating natural language text', Artificial Intelligence, vol. 27, pp. 1-41.
Mackworth, A.K. (1973), 'Interpreting pictures of polyhedral scenes', Artificial Intelligence, vol. 4, pp. 121-137.
Marcus, M.P. (1980), A Theory of Syntactic Recognition for Natural Language, Cambridge, Mass., MIT Press.
Marcus, M.P. (1984), 'Some inadequate theories of human language processing', in T. G. Bever , J. M. Carroll and L. A. Miller (eds), Talking Minds: The Study of Language in Cognitive Science, Cambridge, Mass., MIT Press.
Marr, D. (1976), 'Early processing of visual information', Philosophical Transactions of the Royal Society of London, Series B, vol. 275, pp. 483-525.
Marr, D. (1982), Vision: A Computational Investigation into the Human Representation and Processing of Visual Information, San Francisco, Freeman.
Marr, D. and Hildreth, E. (1980), 'Theory of edge detection', Proceedings of the Royal Society of London, Series B, vol. 207, pp. 187-217.
Marr, D. and Nishihara, H.K. (1978), 'Representation and recognition of the spatial organisation of three-dimensional shapes', Proceedings of the Royal Society of London, Series B, vol. 200, pp. 269-294.
Marr, D. and Poggio, T. (1976), 'Cooperative computation of stereo disparity', Science, vol. 194, pp. 283-287.
Marr, D. and Poggio, T. (1979), ' computational theory of human stereo vision', Proceedings of the Royal Society of London, Series B, vol. 204, pp. 301-328.
Marslen-Wilson, W.D. (1973), 'Linguistic structure and speech shadowing at very short latencies', Nature, vol. 244, pp. 522-523.
Marslen-Wilson, W.D. (1975), 'Sentence perception as an interactive parallel process', Science, vol. 189, pp. 226-228.
259 Mayhew, J.E.W. , and Frisby, J.P. (1981), 'Psychophysical and computational studies towards a theory of human stereopsis', Artificial Intelligence, vol. 17, pp. 349-385.

Mayhew, J.E.W. and Frisby, J.P. (1984), 'Computer vision', in T. O'Shea and M. Eisenstadt (eds), Artificial Intelligence: Tools, Techniques, and Applications, New York: Harper & Row.
Michalski, R. (1983), ' theory and methodology of inductive learning', in R. Michalski , J.G. Carbonell and T.M. Mitchell (eds), Machine Learning: An Artificial Intelligence Approach, Palo Alto, Calif., Tioga Publishing.
Michalski, R.S. and Chilausky, R.L. (1980), 'Learning by being told and learning from examples: an experimental comparison of the two methods of knowledge acquisition in the context of developing an expert system for soybean disease diagnosis', International Journal of Policy Analysis and Information Systems, vol. 4, pp. 125-161.
Michie, D. and Bratko, K. (1978), 'Advice table representations of chess end-game knowledge', Proceedings of the AISB/GI Conference on Artificial Intelligence, pp. 194200.
Minsky, M. (ed.) (1968), Semantic Information Processing, Cambridge, Mass., MIT Press.
Minsky, M. (1975), ' framework for representing knowledge', in P. H. Winston (ed.), The Psychology of Computer Vision, New York, McGraw-Hill.
Mitchell, T.M. (1982), 'Generalization as search', Artificial Intelligence, vol. 18, pp. 203-226.
Mitchell, T.M. Utgoff, P.E. and Baneiji, R. (1983), 'Learning by experimentation: acquiring and refining problem-solving heuristics', in R. Michalski , J.G. Carbonell and T.M. Mitchell (eds), Machine Learning: An Artificial Intelligence Approach, Palo Alto, Calif., Tioga Publishing.
Moore, R.C. (1985), 'Semantical considerations on nonmontonic logic', Artificial Intelligence, vol. 25, pp. 75-94.
Mostow, D.J. (1983), 'Machine transformation of advice into a heuristic search procedure', in R. Michalski , J. G. Carbonell and . M. Mitchell (eds), Machine Learning: An Artificial Intelligence Approach, Palo Alto, Calif., Tioga Publishing.
Newell, A. and Shaw, J.C. (1957), 'Programming the logic theory machine', Proceedings of the Western Joint Computer Conference, vol. 15, pp. 230-240.
Newell, A. and Simon, H.A. (1963), 'GPS, a program that simulates human thought', in E. A. Feigenbaum and J. Feldman (eds), Computers and Thought, New York, McGraw-Hill.
Newell, A. and Simon, H.A. (1972), Human Problem Solving, Englewood Cliffs, New Jersey, Prentice-Hall.
Newell, . , Shaw, J.C. and Simon, H.A. (1957), 'Empirical explorations with the logic theory machine: a case study in heuristics', Proceedings of the Western Joint Computer Conference, vol. 15, pp. 218-230.
260 Nilsson, N.J. (1980), Principles of Artificial Intelligence, Palo Alto, Calif., Tioga Publishing.
Norman, D.A. (1980), 'Copycat science or does the mind really work by table look-up', in R. A. Cole (ed.), Perception and Production of Fluent Speech, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Norman, D.A. , Rumelhart, D.E. and the LNR Research Group (1975), Explorations in Cognition, San Francisco, Freeman.
Norris, D.G. (1982), 'Autonomous processes in comprehension: a reply to Marslen-Wilson and Tyler', Cognition, vol. 11, pp. 97-101.
Papert, S. (1980), Mindstorms: Children, Computers, and Powerful Ideas, New York, Basic Books.
Perrault, C.R. and Allen, J.F. (1980), ' plan-based analysis of indirect speech acts', American Journal of Computational Linguistics, vol. 6, pp. 167-182.
Peters, P.S. and Ritchie, R.W. (1973), 'On the generative power of transformational grammars', Information Sciences, vol. 6, pp. 4983.
Peters, P.S. and Ritchie, R.W. (1982), Phrase Linking Grammars, Department of Linguistics, University of Texas at Austin, Technical Report.
Pitrat, J. (1977), ' chess combination program which uses plans', Artificial Intelligence, vol. 8, pp. 275-321.

Pople, H.E. (1982), 'Heuristic methods for imposing structure on ill-structured problems: the structuring of medical diagnostics', in P. Szolovits (ed.), Artificial Intelligence in Medicine, Boulder, Colorado, Westview Press.
Popper, K.R. (1957), The Logic of Scientific Discovery, London, Macmillan.
Power, R.J. (1979), 'The organization of purposeful dialogues,' Linguistics, vol. 17, pp. 107-152.
Quillian, M.R. (1968), 'Semantic memory', in M. Minsky (ed.), Semantic Information Processing, Cambridge, Mass., MIT Press.
Quillian, M.R. (1969), 'The Teachable Language Comprehender: a simulation program and theory of language', Communications of the Association for Computing Machinery, vol. 12, pp. 459-476.
Ramsay, A. and Barrett, R. (1987), AI in Practice: Examples in POP-11, Chichester, Ellis Horwood.
Raphael, B. (1976), The Thinking Computer: Mind Inside Matter, San Francisco, Freeman.
Rich, C. (1984), 'The programmer's apprentice', in P. H. Winston and K. A. Prendergast (eds), The Al Business: The Commercial Uses of Artificial Intelligence, Cambridge, Mass., MIT Press.
Rich, E. (1983), Artificial Intelligence, New York, McGraw-Hill.
Rieger, C. (1979), 'Five aspects of a full-scale story comprehension model', in . V. Findler (ed.), Associative Networks: Representation and Use of Knowledge by Computers, New York, Academic Press.
261 Ritchie, G.D. (1980), Computational Grammar - An Artificial Intelligence Approach to Linguistic Description , Hassocks, Sussex, Harvester Press.
Ritchie, G.D. and Hanna, F.K. (1984), 'AM: a case study in AI methodology', Artificial Intelligence, vol. 23, pp. 249-268.
Roberts, L.G. (1965), 'Machine perception of three-dimensional solids', in J. T. Tippett , D. A. Berkowitz , L. C. Clapp , C. J. Koester and A. Vanderburgh (eds), Optical and Electro-Optical Information Processing, Cambridge, Mass., MIT Press.
Roberts, R.B. and Goldstein, I.P. (1977), The FRL Primer, Massachusetts Institute of Technology, Laboratory of Artificial Intelligence, Report AIM-408.
Robinson, J.A. (1965), ' machine-oriented logic based on the resolution principle', Journal of the Association for Computing Machinery, vol. 12, pp. 23-41.
Rumelhart, D.E. and Zipser, D. (1985), 'Feature discovery by competitive learning', Cognitive Science, vol 9, pp. 75-112.
Ryle, G. (1949), The Concept of Mind, London, Hutchinson.
Sacerdoti, E.D. (1974), 'Planning in a hierarchy of abstraction spaces', Artificial Intelligence, vol. 5, pp. 115-135.
Sacerdoti, E.D. (1977), A Structure for Plans and Behavior, New York, American Elsevier.
Sacerdoti, E.D. , Fikes, R.E. , Reboh, R. , Sagalowicz, D. , Waldinger, R.J. and Wilber, B.M. (1976), QLISP-A Language for the Interactive Development of Complex Systems, Artificial Intelligence Center, SRI International, Technical Note 120.
Samuel, A.L. (1963), 'Some studies in machine learning using the game of checkers', in E. A. Feigenbaum and J. Feldman (eds), Computers and Thought, New York, McGraw-Hill.
Samuel, A.L. (1970), 'Some studies in machine learning using the game of checkers II: recent progress', in F. J. Crosson (ed.), Human and Artificial Intelligence, New York, Appleton-Century-Crofts.
Schank, R.C. (1972), 'Conceptual dependency: a theory of natural language understanding', Cognitive Psychology, vol. 3, pp. 552-631.
Schank, R.C. (1975), Conceptual Information Processing, Amsterdam, North-Holland.
Schank, R.C. (1982), 'Reminding and memory organization: an introduction to MOPs', in W. G. Lehnert and . H. Ringle (eds), Strategies for Natural Language Processing, Hillsdale, New Jersey, Lawrence Erlbaum Associates.

Schank, R.C. and Abelson, R.P. (1977), Scripts, Plans, Goals and Understanding , Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Schank, R.C. and Colby, K.M. (eds) (1973), Computer Models of Thought and Language, San Francisco, Freeman.
Schubert, L.K. , Goebel, R.G. and Cercone, N.J. (1979), 'The structure and organisation of a semantic net for comprehension and inference', in . V. Findler (ed.), Associative Networks: Representation and Use of Knowledge by Computers, New York, Academic Press.
262 Searle, J.R. (1969), Speech Acts: An Essay in the Philosophy of Language, Cambridge, Cambridge University Press.
Searle, J.R. (1980), 'Minds, brains, and programs', Behavioral and Brain Sciences, vol. 3, pp. 417424.
Searle, J.R. (1984), Minds, Brains and Science, London, BBC.
Selfridge, M. (1986), ' computer model of child language learning', Artificial Intelligence, vol. 29, pp. 171-216.
Shannon, C. (1950), 'Programming a digital computer for playing chess', Philosophy Magazine, vol. 41, pp. 356-375.
Shirai, Y. (1973), ' context-sensitive line finder for recognition of polyhedra', Artificial Intelligence, vol. 4, pp. 95-120.
Shortliffe, E.H. (1976), MYCIN: Computer-based Medical Consultations, New York, American Elsevier.
Shortliffe, E.H. and Buchanan, B.G. (1979), ' model of inexact reasoning in medicine', Mathematical Biosciences, vol. 23, pp. 361-379.
Sidner, C.L. (1983), 'Focusing in the comprehension of definite anaphora', in M. Brady and R. C. Berwick (eds), Computational Models of Discourse, Cambridge, Mass., MIT Press.
Sussman, G.J. (1975), A Computer Model of Skill Acquisition, New York, American Elsevier.
Sussman, G.J. , Winograd, T. and Charniak, E. (1971), Micro-Planner Reference Manual, Massachusetts Institute of Technology, Laboratory of Artificial Intelligence, Memo no. 203a.
Thorne, J.P. , Bratley, P. and Dewar, H. (1968), 'The syntactic analysis of English by machine', in D. Michie (ed.), Machine Intelligence 3, Edinburgh, Edinburgh University Press.
Torrance, S. (1986), 'Breaking out of the Chinese room', in M. Yazdani (ed.), Artificial Intelligence: Principles and Applications, London, Chapman & Hall.
Turing, A.M. (1936), 'On computable numbers, with an application to the Entscheidungsproblem', Proceedings of the London Mathematical Society, Series 2, vol. 42, pp. 230-265.
Turing, A.M. (1950), 'Computing machinery and intelligence', Mind, vol. 59, pp. 433-460.
Ullman, S. (1979), The Interpretation of Visual Motion, Cambridge, Mass., MIT Press.
Waltz, D. (1975), 'Understanding line drawings of scenes with shadows', in P. H. Winston (ed.), The Psychology of Computer Vision, New York, McGraw-Hill.
Wanner, E. (1980), 'The ATN and the sausage machine: which one is baloney?', Cognition, vol. 8, pp. 209-225.
263 Wanner, E. and Maratsos, M. (1978), 'An ATN approach to comprehension', in M. Halle , J. Bresnan and G. A. Miller (eds), Linguistic Theory and Psychological Reality, Cambridge, Mass., MIT Press.
Webber, B.L. (1983) 'So what can we talk about now?', in M. Brady and R. C. Berwick (eds), Computational Models of Discourse, Cambridge, Mass., MIT Press.
Weiss, S.M. , Kulikowski, C.A. , Amarai, S. and Saffir, A. (1978), model-based method for computer-aided medical decision-making', Artificial Intelligence, vol. 11, pp. 145-172.
Weizenbaum, J. (1966), 'ELIZA - a computer program for the study of natural language communication between man and machine', Communications of the Association for Computing Machine, vol. 9, pp. 36-45.

Weizenbaum, J. (1976), Computer Power and Human Reason, San Francisco, Freeman.
Whitehead, A.N. and Russell, B. (1910), Principia Mathematica, vol. I, Cambridge, Cambridge University Press.
Wilensky, R. (1982), 'Points: a theory of the structure of stories in memory', in W. G. Lehnert and . H. Ringle (eds), Strategies for Natural Language Processing, Hillsdale, New Jersey, Lawrence Erlbaum Associates.
Wilensky, R. (1983), Planning and Understanding, Reading, Mass., Addison-Wesley.
Wilkins, D.E. (1979), 'Using plans in chess', Proceedings of the Sixth International Joint Conference on Artificial Intelligence, pp. 960-967.
Wilkins, D.E. (1984), 'Domain-independent planning: representation and plan generation', Artificial Intelligence, vol. 22, pp. 269-301.
Wilks, Y. (1975), ' preferential, pattern-seeking semantics for natural language interface', Artificial Intelligence, vol. 6, pp. 5374.
Winograd, T. (1972), Understanding Natural Language, New York, Academic Press and Edinburgh, Edinburgh University Press (Also published in Cognitive Psychology, vol. 3, pp. 1-191.)
Winograd, T. (1973), ' procedural model of language understanding', in R. C. Schank and . M. Colby (eds), Computer Models of Thought and Language, San Francisco, Freeman.
Winograd, T. (1975), 'Frame representations and the declarative/procedural controversy', in D. G. Bobrow and A. M. Collins (eds), Representation and Understanding, New York, Academic Press.
Winograd, T. (1983), Language as a Cognitive Process, vol. 1: Syntax, Reading, Mass., Addison-Wesley.
Winston, P.H. (ed.) (1975a), The Psychology of Computer Vision, New York, McGraw-Hill.
Winston, P.H. (1975b), 'Learning structural descriptions from examples', in P. H. Winston (ed.), The Psychology of Computer Vision, New York, McGraw-Hill.
264 Winston, P.H. (1984), Artificial Intelligence, 2nd edn, Reading, Mass., Addison-W esley.
Winston, P.H. and Horn, B.K.P. (1984), LISP, 2nd edn, Reading, Mass., Addison-Wesley.
Wolf, J. and Woods, W.A. (1980), 'The HWIM speech understanding system', in W. A. Lea (ed.), Trends in Speech Recognition, Englewood Cliffs, New Jersey, Prentice-Hall.
Woodham, R.J. (1981), 'Analysing images of curved surfaces', Artificial Intelligence, vol. 17, pp. 117-140.
Woods, W.A. (1970), 'Transition network grammars for natural language analysis', Communications of the Association for Computing Machinery, vol. 13, pp. 591-606.
Woods, W.A. (1975), 'What's in a link? Foundations for semantic networks', in D. G. Bobrow and A. M. Collins (eds), Representation and Understanding, New York, Academic Press.
Woods, W.A. (1977), 'Lunar rocks in natural English: explorations in natural language question answering', in A. Zampolli (ed.), Linguistic Structures Processing, Amsterdam, North-Holland.
Woods, W.A. (1981), 'Procedural semantics', in A. K. Joshi , B. L. Webber and I. A. Sag (eds), Elements of Discourse Understanding, Cambridge, Cambridge University Press.
Younger, D.H. (1967), 'Recognition and parsing of context-free languages in time n3 ', Information and Control, vol. 10, pp. 189-208.