An Introduction to Expert Systems

Michel Gondran

An Introduction to Expert Systems

Michel Gondran
Translated from the French by
Joanna Gosling
English edition prepared by
Peter Gosling

McGRAW-HILL Book Company (UK) Limited

London · New York · St Louis · San Francisco · Auckland · Bogotá
Guatemala · Hamburg · Johannesburg · Lisbon · Madrid
Mexico · Montreal · New Delhi · Panama · Paris · San Juan
São Paulo · Singapore · Sydney · Tokyo · Toronto

Published by
McGRAW-HILL Book Company (UK) Limited
MAIDENHEAD · BERKSHIRE · ENGLAND

British Library Cataloguing in Publication Data
Gondran, Michel
An introduction to expert systems.
1. Expert systems (Computer science)
I. Title II. Introduction aux systèmes expertes. *English*
006.3'3 QA76.9.E96

ISBN 0-07-084920-X

Library of Congress Cataloging-in-Publication Data
Gondran, Michel.
An introduction to expert systems.

Translation of: Introduction aux systèmes experts.
Bibliography: p.
1. Expert systems (Computer science) I. Title.
QA76.76.E95G6613 1986 006.3'3 85-23696
ISBN 0-07-084920-X

First published by Edition Eyrolles, Paris
First edition © Edition Eyrolles, 1983

12345 WL 89876

Typeset by Eta Services (Typesetters) Ltd., Beccles, Suffolk and printed and bound in Great Britain by Whitstable Litho Limited, Whitstable, Kent

An Introduction to Expert Systems

Acknowledgement

Grateful thanks to Kevin Hughes, Simon Parry, and Mike Barrett of Expertech Research Limited for their help in the preparation of the English edition.

Contents

1. Introduction

The goal of an artificial intelligence system is to analyse human behaviour in the fields of perception, comprehension, and decision making in the ultimate hope of reproducing them on a machine, namely a computer.

An expert system is the first true operational application of this type of research and this book aims to present the basic structure of an expert system in a simple form and to point towards its future development.

As an introduction we will specify the place of expert systems in the field of artificial intelligence. In Chapter 2 we will describe the general principles behind such systems by means of a few examples which serve as their foundation.

Chapter 3 is a short introduction to the role of artificial knowledge in the development of expert systems and in Chapter 4 we present an objective account of the SNARK language developed by J. L. Laurière. It would appear that SNARK currently forms the basis of many projected expert systems. Four examples are presented in this chapter as illustrations of the use of this technique.

Then in Chapter 5 the algorithm for an expert system is developed in more detail, while Chapter 6 describes several applications of expert systems.

Chapter 7 is devoted to a survey of the future role of expert systems.

1.1 Artificial intelligence

The study of artificial intelligence (AI) concerns all those activities of man for which there are no set procedures laid down. If computer science is the science of information processing by a set algorithmic approach then the study of AI looks at all those activities which cannot be treated at a simple, standard, algorithmic level.

There are so many activities, even in trivial situations such as reading a book or an article, where what we do and how we do it is not governed by a set of well-defined rules. For example, a character such as '!' can be interpreted, according to its context, by our visual system sometimes as an 'i', sometimes as an 'l' or even as a specially defined character. In the case of handwriting things are even worse. The following three words all start with

the same graphic symbol *n*, but because of its context it becomes endowed with three quite different meanings: *notive*, *note*, *ntility*. How does our cognitive system remove these ambiguities? This is just one of the problems highlighted in the study of AI.

The traditional programming languages do not allow us to communicate with the computer in anything but an algorithmic manner by issuing a series of orders. In AI, heuristics, or rules of thumb, are substituted for algorithms.

These rules of thumb should take account of the characteristics of the particular problem being studied. They consider not only the data but also its context.

Let us consider the way we perform the task of adding two numbers together. If the numbers are very small, say less than 20, we will often obtain the result by 'fusing' the two numbers together. If the numbers are larger than about 20 then we may have to perform a more complicated operation using 'carrying' figures in our head. If the numbers are too big to add together in this way then we may have to write the entire sum down on paper. This is a case where we do not always carry out the operation in the same manner every time. The method we employ depends on the data we are handling.

The earliest research into AI often had goals which were overambitious. The researchers could see that what they were doing had applications in a vast number of different areas, but their hopes rested on an underestimation of the size of the problems which had first of all to be solved before a practical system could be evolved. Automatic language translation programs were an early example of this situation.

Work on automatic translation began in the fifties. Each word in a group had a set of alternative translations and simple rules were established in order to put the appropriate translated words into a sensible order and to improve syntax. This highly syntactic approach produced some terrible blunders. For example, when the English sentence 'The spirit is willing, but the flesh is weak' was translated into Russian and then back into English again it emerged as 'Vodka is strong, but meat is rotten'.

Work on artificial translation was almost totally abandoned in the mid-sixties as the result of the ALPAC report which had been commissioned to evaluate the project. The report came to the conclusion that a suitable system was impossible to devise.

You have therefore to go via the meaning of the text in order to be able to translate it properly and you must not be taken in, as so many people were in 1966, by the ELIZA program of J. Weizenbaum. This program short-circuits real linguistic processes and uses a clever system of models with fixed answers which mimic language in a very convincing way. The program responses imi-

tate those of a psychiatrist. Each response is drawn from a series of sentences or phrases which are kept in the memory and are triggered off by certain words, or words belonging to a certain group, which appear in the sentences typed in by the 'patient'. For example, each time the word 'mother' is mentioned the program produces the response of the following type, 'Tell me about your mother'. If you type in 'I feel a little tired', ELIZA uses part of the patient's sentence in its reply, 'Why are you feeling a little tired?'.

Weizenbaum realized, as a result, that this program showed how dangerous it was to take the simulation of human behaviour as a gauge with which to measure the 'intelligence' of a computer. He noticed that ELIZA operated at a very simple and superficial level of language comprehension and that, in spite of this, some people were fooled by the apparent reality of the responses to the point where they were confiding their personal problems to the computer as if it really were a psychiatrist.

The most important developments in AI were made on problem solving and were centred on the intelligent discovery of solutions thanks to the rules of thumb which had been given to the machine. As examples, see the GENERAL PROBLEM SOLVER program by A. Newall and H. Simon (Nobel Prize for Economics, 1978) and the ALICE program of J. L. Laurière (European Research Prize CII-HB, 1982).

1.2 Expert systems

For several years now researchers have noticed that one essential element was missing from all the existing programs. It was that there seemed to be a lack of in-depth knowledge of the area which is being dealt with at any time. In other words, there was none of the cumulative knowledge which comes from years of practice and experience. This cumulative knowledge comes from making analogies, relating pieces of information, and possessing significant amounts of knowledge in a wide variety of fields.

Information which is simply stored in a database is easy to extract. What is difficult is to deduce the facts which have not been recorded in the database but could be deduced from the stored information.

The aim of an expert system is to reproduce the behaviour of a human expert, thus performing an intellectual task in a specific field. Expert systems position themselves at the junction of the two approaches to AI; the representation of information on the one hand and its automatic 'demonstration' on the other.

These form two totally independent systems:

- a knowledge base,
- a 'demonstrator' of theorems (an 'inference engine' or rule interpreter).

The knowledge base translates expert knowledge in a given field into a declaratory and modular form. The 'demonstrator' has the task of calling up and using this information in a meaningful way in order to answer a question or solve a problem.

An expert system differs from a conventional computer program in two essential ways since at any time an expert system can:

- explain its behaviour to the human expert,
- receive new pieces of information from the human expert without any new programming being required.

The knowledge base must be readable on its own and must exist independently from the 'inference engine', but must be capable of being interpreted by it, and this knowledge base must be under the control of the human expert.

2. How to structure the base of an expert system

2.1 Schema of an expert system

Language is the most general current representation of information. It may be broken down into sentences which form blocks or modules of information. Expert systems use this modular approach to the structure of information.

In the case of most expert systems the information used will be one of two types. One is a base of facts and the other is a base of rules of production of these facts. It is this type of expert system we will deal with in this book. Other representations are covered by Schank (1977), Charniak (1978), Codier (1979), and Pastre (1978).

Example 1

Let us consider the following knowledge base:

Initial base of facts: H,K

Base of rules: (R1) $A \rightarrow E$
(R2) $B \rightarrow D$
(R3) $H \rightarrow A$
(R4) E and $G \rightarrow C$
(R5) E and $K \rightarrow B$
(R6) D and E and $K \rightarrow C$
(R7) G and K and $F \rightarrow A$

where $A \rightarrow E$ means 'E may be deduced from A'.

The inference engine induces from the facts contained in the knowledge base and uses the rule of *modus ponens* which is the rule that 'if P is true and if $P \rightarrow Q$ then Q must be true'.

Two general strategies of demonstration are therefore possible:

1. *Forward chaining* Start from the original base of facts and trigger off the rules for which the premises on the left-hand side are satisfied. Add to this the facts which are obtained in this way and continue until 'saturation point' is reached.

In Example 1 a chain is obtained from the following derivation:

$H \rightarrow A$ (R3)	(base of facts =	$\{A, H, K\}$
$A \rightarrow E$ (R1)		$\{A, E, H, K\}$
E and $K \rightarrow B$ (R5)		$\{A, B, E, H, K\}$
$B \rightarrow D$ (R2)		$\{A, B, D, E, H, K\}$
D and E and $K \rightarrow C$ (R6)		$\{A, B, C, D, E, H, K\}$

2. *Backward chaining* If your aim is to show that fact D is correct then you look at all the rules which have that aim on the right-hand side. Each of these rules is then considered and if all the premises have been satisfied in the original base of facts then you have reached your goal; otherwise the unknown premises are recorded as if they were new aims and you begin the cycle again for each of them.

In Example 1 the rule R2 is considered and B is defined as a new goal. Then rule R5 is considered and E is defined as a new goal. Then rule R1 is considered and A is defined as a new goal. Then rules R3 and R7 are considered and it is R3 which provides the conclusion. We have been led backwards through rules R2, R5, R1, and R3 to the two things we do know, which are H and K. The pursuit of a goal is usually represented by means of an AND/OR tree, and in this case the tree would be as shown in Fig. 2.1.

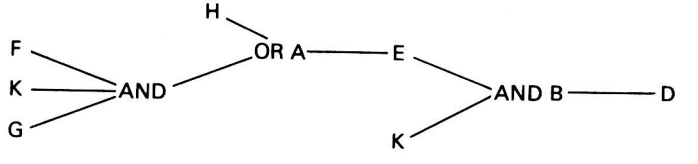


Fig. 2.1

Note 1: The forward chaining strategy is used in the SNARK inference engine (Laurière, 1982c), and the backward chaining in the EMYCIN inference engine (Van Melle, 1979).

Example 2

This comes from the field of botany and we will look at the base of rules which follow. In this the proposition 'plants have a flower' is abbreviated to 'flower'. The $\wedge$ symbol corresponds to the logical connector 'and' and the $\neg$ symbol corresponds to negation.

(a) IF flower $\wedge$ seed THEN phanerogam
(b) IF phanerogam $\wedge$ unprotected seed THEN fir tree

(c) IF phanerogam ∧ 1-cotyledon THEN monocotyledon
(d) IF phanerogam ∧ 2-cotyledons THEN dicotyledon
(e) IF monocotyledon ∧ rhizome THEN lily of the valley
(f) IF dicotyledon THEN anemone
(g) IF monocotyledon ∧ (¬ rhizome) THEN lilac
(h) IF leaf ∧ flower THEN cryptogam
(i) IF cryptogam ∧ (¬ root) THEN moss
(j) IF cryptogam ∧ root THEN fern
(k) IF (¬ leaves) ∧ plant THEN thallophyte
(l) IF thallophyte ∧ chlorophyll THEN algae
(m) IF thallophyte ∧ (¬ chlorophyll) THEN mushroom
(n) IF (¬ leaf) ∧ (¬ flower) ∧ (¬ plant) THEN colon bacillus

Thus (a) means that 'IF the plant has a flower and a seed THEN the plant has a phanerogam'.

The problem we are faced with is to determine what the plant is which has the following characteristics: rhizome, flower, seed, 1-cotyledon. This forms our current base of facts. If we use forward chaining we get the following derivation:

(a) → phanerogam
(c) → monocotyledon
(e) → lily of the valley

and these are the only rules we use in this case to obtain the solution 'lily of the valley'.

As a result the facts and the rules of production must be more closely defined.

Fundamentally, two types of production rules are used. One is based on propositional logic and the other on first-order logic.

2.2 First-order logic

Examples 1 and 2 represent typical cases of propositional logic. The rules apply directly to the facts, i.e., the premises and consequences of the rules are explicitly the facts. On the other hand, the rules which are based on first-order logic can use variables and quantifiers and can also be applied to a whole class of facts.

As a result, rules may be introduced such as:

IF man (x) THEN mortal (x)

which means: 'If x is a man THEN x is a mortal'. The well-known syllogism which is generally attributed to Aristotle:

All men are mortals,
Socrates is a man,
Therefore Socrates is mortal.

which can be described as follows:

The initial base of facts: man (Socrates)
(which means 'Socrates is a man')

The initial base of rules: IF man (x) THEN mortal (x)
(which means 'men are mortal')

Therefore the following fact may be deduced:
mortal (Socrates)

You will notice that, in this case, in order to apply the rule you must look at the base of facts for a man which will allow you to particularize the variable x. Another interesting example is the possibility of using theorems such as those of transitivity which can be represented by the following rules:

IF element (y) = x
and element (z) = y
THEN element (z) = x

(which means 'IF x is an element of y and if y is an element of z THEN x is an element of z').

So, from the fact that the tyre is an element of the wheel (element (wheel) = tyre) and that the wheel is an element of the bicycle (element (bicycle) = wheel), the rule of transitivity allows the system the knowledge that the tyre is an element of the bicycle (element (bicycle) = tyre).

This very general rule allows us to state an important piece of information in an implicit way.

Another example of a rule of first-order logic is:

IF father (x) = y
  mother (y) = z
THEN grandmother (x) = z

(which means 'if y is the father of x and z is the mother of y, then z is the grandmother of x').

Here are two more examples taken from *La Logigue sans peine* by Lewis Carroll (*Symbolic Logic*, 1896) which illustrate a number of useful points.

Example 3

1. Babies are illogical.
2. Never despise the man who can stand at arm's length from a crocodile.
3. Illogical people are despised.

Come to a conclusion about these statements.

Let us translate these three statements into the language of first-order logic.

1. ¬ logical (babies)
2. IF stand at arm's length from a crocodile (x) THEN ¬ despised (x)
3. IF ¬ logical (x) THEN despised (x)

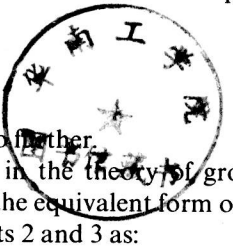
(where statement 3, for example, means 'If x is illogical, then x is despised').

Statement 1 makes up the base of facts. Statements 2 and 3 make up the base of rules.

Using statements 1 and 3 we can deduce that

despised (babies)

which mean 'babies are despised'. Then we can go no further.



However, there is a classic result which occurs in the theory of groups which allows us to write the rule 'IF A THEN B' in the equivalent form of 'IF B THEN A'. This means that we can write statements 2 and 3 as:

2a. IF despised (x) THEN ¬ stand at arm's length from a crocodile (x)
3a. IF ¬ despised (x) THEN logical (x)

The base of rules is now 2, 2a, 3, and 3a. In using 1, 3, and 2a in succession we may come to the conclusion that:

¬ stand at arm's length from a crocodile (babies)

(which means 'babies cannot stand at arm's length from crocodiles').

Note 2: Even in a simple example the transformation into first-order logic is not completely 'trivial'. In order to be able to use forward chaining for deductions we are forced in this case to represent each statement in one of two ways (2 and 2a, 3 and 3a).

Example 4

1. Animals are always mortally offended if I take no notice of them.
2. The only animals which belong to me are in this meadow.
3. No animal is able to solve a riddle if he has received no suitable education.
4. None of the animals in this meadow are water rats.
5. When an animal is mortally offended it always starts to run about and scream.

6. I never pay attention to an animal which does not belong to me.
7. No animal which has received suitable education starts to run about and scream.

Find a conclusion to these seven statements.

First of all we will translate these statements using first-order logic. The variable x in this case applies to 'animal'.

1. IF ¬ attention (x) THEN mortally offended (x)
2. IF ¬ meadow (x) THEN ¬ belongs to (x)
3. IF ¬ education (x) THEN ¬ solve riddle (x)
4. ¬ meadow (water rat)
5. IF mortally offended (x) THEN runs and screams (x)
6. IF ¬ belongs to (x) THEN ¬ attention (x)
7. IF runs and screams (x) THEN ¬ education (x)

Statement 4 makes up the base of facts. The others make up the base of rules.

We can make a simple deduction by using statements 4, 2, 6, 1, 5, 7, and 3 in that order in order to deduce that 'no water rat is able to solve a riddle'.

Note 3: You can already notice a slight difference between databases. In the case of the databases used in data processing it is the knowledge of facts which represents virtually the sum total of the knowledge. With expert systems it is the base of rules which represent virtually the sum total of the knowledge. In the case of a diagnosis problem, the base of facts is only concerned with the data which is relative to the current situation. For a medical diagnosis this would be the data which is provided by a particular patient. For the diagnosis of an 'incident' the data would have been provided by different witnesses.

The use of propositions in expert systems does not lend itself to constructing wide-ranging and extensive systems.

Systems of the future will be based on first-order logic, cf. OPS, SNARK (Laurière, 1982c), PROLOG (Colmerauer, 1977), TANGO (Cordier and Rousset, 1982), ALOUETTE (Mulet-Marquis, 1983).

2.3 The inference engine

Let us consider the case of forward chaining. Our inference engine knows on the one hand the order in which it will try to apply the rules and on the other hand the order in which they will actually be applied.

In the case of most existing expert systems (cf. PROSPECTOR systems (Konolige, 1979), SAM (Gascuel, 1981), and the systems based on PRO-