

Book Reviews

Expert systems in business—a practical approach by M L Barratt and A C Beerel. Reviewed by Peter Sell.

Interpreting anaphors in natural language text by David Carter. Reviewed by John Beaven.

Developing expert systems for business by Benchimol, Levine and Pomerol. Reviewed by N A D Connell.

Automated theorem proving (second revised edition) by Wolfgang Bibel. Reviewed by G A Ringwood.

Expert systems in business—a practical approach by M L Barratt and A C Beerel, Ellis Horwood Ltd, Chichester 1988, pp 259, £29.25.

Over the past three years the number of books on expert systems has grown from a handful to a small library. This abundance is, however, mostly descriptive and conceptual; few offer practical help borne of experience or rise above platitudes and vague generalities. Mike Barrett and Annabel Beerel's book, *Expert Systems in Business*, is one of these few.

The book's aim is to help those as yet inexperienced in building expert systems to identify applications of this technology and to implement systems of clear business benefit. The authors can call upon several years of experience in building just such systems which, when coupled with their down to earth style and abundance of examples, has allowed them to create a book that is not only useful but also a pleasure to read.

The book gives a detailed account of knowledge engineering in practice, covering in detail aspects of identifying practicable and worthwhile applications, planning and managing the projects, using experts, codifying know-how, and proceeding in use of this technology in a business-like, hype- and mysticism-free manner, from inception to economic or organizational gains.

The book is divided into six parts which deal with the introduction to the technology, finding profitable applications, the inner workings of these systems, hardware and software tools, how to set about building applications, and a final summary of the key points.

It is primarily a practitioner's book and of greatest potential benefit to the working knowledge engineer, but the book covers material of vital interest to managers and planners. Unfortunately, perhaps because of this broad coverage, information and advice particularly important to managers is interleaved with the more technical detail. This, I fear, will put off some—in many cases those who would benefit most—from reading it. It really should not, however! Not only are even the most technical topics presented in a clear and easy to digest manner, but the authors never lose sight of their aim to present information relevant to practical needs and from a common sense standpoint.

Each part is peppered with nuggets of practical wisdom, often expressed in the form of simple heuristics. For example, this is what the authors recommend on choosing software:

Use a shell if you can, an environment where you should, and an AI language when you must.

Or on how to deal with information casually offered by experts:

Nothing is ever "by the way"

The greatest danger the authors exposed themselves to is being accused of stating the obvious. Unfortunately, the laws of libel prevent quoting the numerous failures due to ignoring "the obvious". One of the guiding principles they give is:

It's never too early to show the system to users.

This is the advice probably most frequently ignored; which accounts for an astonishing number of finished or nearly finished expert systems that were actively sabotaged or just simply ignored by the intended users. Obvious it may be, but ignoring it leads to waste of money and missed opportunities.

There are also numerous checklists to help you make choices, to keep you out of trouble, or provide you with a fresh approach when you are stuck.

In brief, I would thoroughly recommend this book to aspiring or practising knowledge engineers and their managers in organizations that wish to exploit expert system technology for gain. If, however, you seek for theoretical niceties, this book is not for you.

Reviewed by Dr Peter Sell, Digital Equipment Co Ltd, PO Box 121, Reading, Berkshire RG2 0TE.

Interpreting anaphors in natural language text by David Carter, Ellis Horwood series in Artificial Intelligence 1987, pp 290, \$39.90, ISBN 0-7458-0250-8.

- If I died, would you marry again?
- Yes.
- Would you let her live in my house?
- Yes.
- And would she sleep in my bed?
- Yes.
- And drive my car?
- Yes.
- And use my golf clubs?
- No, definitely not.
- Why not?
- Because she's left-handed.

Anaphora resolution consists of deciding what it is, in a given discourse, that pronouns (words like *it*, *him*, *her*) and other such expressions refer to. Understanding how anaphora works is a key issue in any natural language processing application, such as NL understanding, or the generation of readable text.

Anaphora resolution in its full generality is still an open research topic. It is clear, however, that the knowledge that must come into play involves syntax, semantics, discourse phenomena such as focus, as well as much world knowledge. For instance, in a sentence like:

John asked Bill to shave him

the syntax of English prevents *him* from referring to *Bill*. In general, however, the issue is not so simple, as in:

John dropped the hammer on Bill's foot. He howled in pain.

As representing major pieces of world knowledge and doing inference with them is still beyond the state-of-the-art, the question arises as to whether other more manageable pieces of knowledge may be successfully used for anaphora resolution.

In this book, which is closely based on his Ph.D thesis, David Carter demonstrates to what extent a "shallow processing" approach, in which linguistic knowledge is exploited as much as possible in order to minimize the need for world knowledge, can yield satisfactory results, and where the "shallow processing" fails.

The book integrates several recent theories, such as Wilks' Preference Semantics and Sidner's theories about focus (both of which are modified to some extent), into a program called SPAR (Shallow Processing Anaphor Resolver). SPAR resolves anaphoras in simple English stories, and produces paraphrases to show how they were resolved. Of course, the paraphrases are quite unreadable, because all ambiguities are expanded in full (that is why we use anaphora in the first place!), but the point of the exercise is to disambiguate the text.

The book starts with a fairly comprehensive classification of anaphoric phenomena in English, and with a review of the main pieces of work that are relevant to SPAR: Wilk's Preference Semantics and the inference theory associated with it, Boguraev's analyzer, and Grosz and Sidner's theor-