

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-

ies on focusing and discourse structure. The author then explains on the main part of the book how these theories are integrated, and how inference and generation mechanisms are added into SPAR.

Although the book reads like a good Ph.D thesis, I feel that the publisher's referees should have exercised a greater editorial power in order to turn it into a more readable book. In particular, the initial section should have been greatly expanded to make it more accessible to its intended wide readership, which according to the cover consists of those in "artificial intelligence, computational linguistics, cognitive science, pragmatics, computer science, English language, psycholinguistics, linguistics, expert systems and knowledge engineering". The references to the ideas on which SPAR is based may be sufficient to refresh the memory of the working computational linguist who is already familiar with them, but I do not think they are sufficient for the reader who meets them for the first time. It is unfortunate therefore that the whole work seems somewhat rushed, which is probably a reflection on the pressure that publishers are under to get books out into the market in a rapidly moving field.

Reviewed by John Beaven, Department of Artificial Intelligence, University of Edinburgh, 80 South Bridge, Edinburgh EH1 1HN.

Developing expert systems for business by Benchimol, Levine and Pomerol, North Oxford Academic Press, 1987.

This book is intended to introduce a non-technical audience to some of the technical and managerial aspects of expert systems. It takes the stance of providing information for the potential users of such systems, claiming therefore to be neither too theoretical nor too superficial. It is written in an uncomplicated style, which is translated from an original French language edition published in 1986.

In its intent, level and style it most closely resembles Harmon and King's *Expert Systems* (Wiley, 1985), which is similarly aimed at the business manager. It sets out to demonstrate the role expert systems might make in business decision making, with an emphasis very much on promise rather than fulfillment, for as the authors honestly acknowledge in the book's foreword, "there are in fact very few (expert systems) operational in businesses". This, the authors conclude, is because insufficient attention is paid to the feasibility of the applications chosen, and a lack of careful preparation in their implementation. The final quarter of the book is devoted to some of the factors surrounding the introduction of expert systems into an organization, which the authors claim, with some justification, are often given insufficient prominence in the literature.

Chapters 1 to 6 define expert systems and their operations. All very clear and mostly to the point, although there is little in these chapters to distinguish the book from many similar "management appreciation" expert system books. Chapter 7 considers the relationship between AI, DSS and expert systems, but does not really do justice to its subject matter. The chapter is a mere six pages long and perhaps it would have been better either to incorporate this material elsewhere or to examine its relevance to the business development of expert systems in greater depth. Chapter 8 looks at shells and hardware. As might be expected, such an examination becomes quickly dated. Of the overview presented in the section entitled "recent systems", the criteria for inclusion is not made clear, and the functionality for business applications of each shell, which in view of the book's readership is likely to be of particular interest, is given insufficient prominence. The choice of shells has a particularly French flavour; some old favourites familiar to UK readers are not included (SAGE, SAVOIR) and the passage of time since the book's original publication has meant that most of the newer micro-based shells are not considered.

Chapters 9, 10 and 11 concentrate on some of the wider organizational issues referred to above. The authors suggest that implementors of expert systems ask themselves a number of questions before starting, along the lines of who, what, where, how, when and how much? An excellent framework, but sadly omitting what could be the most crucial business question, why?

In providing guidance to help the potential business user select his first application, the authors note a useful distinction between selection criteria associated with the application, the organization,