The Knowledge Engineering Review, Vol. 9:4, 1994, 421–424

Book reviews

Simply logical—intelligent reasoning by example by Peter Flach, John Wiley, Chichester, 1994, pp 240, £19.95, ISBN 0471 94152 2.

This book is neatly divided into three sections, each comprising three chapters. It is not clear whether this was by chance or design; the partitioning is quite natural since each section is self-contained.

The whole of the first section is an introductory text on computational logic, some of its theoretical bases and its realization in the Prolog programming language. The middle section looks at some specific AI representation and reasoning techniques, namely searching through directed graphs, from a logic programming perspective. The final section deals with two contemporary issues in logic programming research; and interpretation of natural language; and reasoning with incomplete information. The subject matter of most chapters is covered at some pace, with hardly a moment for the reader to pause and catch breath.

For a practitioner who had forgotten the theoretical basis of logic programming, chapter 2 was an excellent reminder, if somewhat daunting. However, the techniques introduced in the middle section and the research areas of the final section will interest only an elite collection of practitioners.

The Prolog programming techniques advocated in chapter 3 of the introductory section are rather trivial. Worse still, several of the following Prolog programs are badly constructed with many inefficiencies. This book is definitely not for the commercial/industrial Prolog practitioner.

For the informed researcher, much of the material in the first two sections is well known and understood. The research material of most interest appears in the final section, especially the discussions of abduction and induction of logical explanations.

For a mere 40 pages of contemporary material, though, the book is quite expensive. Far better, and probably more fruitful, to search out the original research papers upon which those final chapters are based.

The feeling throughout this book is that it was written from a set of course notes. As such, it is primarily for undergraduate or masters students taking a broad range of courses in logic programming and AI.

Reviewed by Dr Philip Vasey, Logic Programming Associates Ltd., London, UK

Text generation—using discourse strategies and focus constraints to generate natural language text by Kathleen R. McKeown, Cambridge University Press, 1992, pp 246, £13.95, ISBN 0-521-43802-0.

Text generation, or text production, is not the study of what academics do to justify their existence, but rather an area of language engineering that has been waiting to happen for quite some time. Text production is usually motivated by the goal of generating responses to a query system, often some expert system, in the form of several interconnected sentences. It is the other side of the coin to text interpretation, which is currently enjoying a great deal of attention, and although some aspects of the production and interpretation processes are shared, they are different enough to have quite distinct lineages in the literature.

A frequent scenario for a text production system would be that the user has made a query (possibly related to earlier queries), and some processing has resulted in a solution being found—

represented internally to that processing system. It is the job of the text production system to convey the internally represented answer to a human reader in an appropriate manner. And it is this choice of what is appropriate that consumes the energies of researchers: what is the most appropriate sentence structure to use; which word best captures the concept the system trying to impart; what has been said before, or can be assumed; and before too long you are looking at intention, motivation, and so on.

Kathleen McKeown's book describes the text generation method used in a system called TEXT, which responds to queries made to a military database. The basis of structuring texts in TEXT is schemata, derived from the psychological-cognitive approach to text understanding. As such, TEXT, is often considered the first comprehensive system of its kind. Which brings us to an important issue for prospective readers of this book. The first publication of this book was 1985, nearly a decade ago, and McKeown has developed (and written about) more sophisticated text production systems since then (the COMET system in 1990). Most of the references in the book are before 1980, and consequently there is a musty air to the whole description of the work. So, it is a work that is really only of academic interest now, and this does rather beg the question of why it was republished recently (1992) in its original form. Reprinting it in paperback form rather suggests that the imprint is being aimed at a wider audience than the original—students, perhaps. In which case, I think the least that a publisher should do is provide the author with the chance to add a foreword or additional section that highlights where the work sits in current research (I can't believe that an author would turn down the opportunity to provide references to their more recent work).

The book itself is very clearly written, with a pleasant mix of theoretical background and technical detail; it is a model for a postdoctoral thesis. The weaknesses of the TEXT system, such as its limited inference facilities and hand-crafted knowledge base, are discussed as well as its innovative (at the time) aspects. The content of the database with which the system operates may not be to everyones taste—it can be disconcerting to be faced with knowledge base descriptions such as "FREE-FALLING DESTRUCTIVE-DEVICE (non-restrictive FUNCTION LETHAL-KILL)" in our peace-loving (but not always peace-doing) 1990s. Essentially, the book has three sections, about 80 pages addressing introductory and theoretical issues of discourse structure and focus constraints, followed by about the same number of pages describing the implementation of the TEXT system. The final 50 pages cover discourse history—how responses can be improved in the light of previous responses—related work and sample program output.

Elements of the implementation are a bit sad (but honestly reported), such as the way in which queries are made. A user makes queries by calling one of three LISP functions: information, definition and differense (the latter spelled so that it doesn't conflict with the built-in LISP function). However, in the context of a historical piece of work, this should not put the prospective reader off. The essential idea that TEXT embodies is that text production needs to be able to make use of discourse and focus in order to generate coherent responses, and these issues still apply today. The fact that researchers active in this field still refer to this book and the TEXT system indicates that it is a stepping stone in the evolution of text production systems. As such, this book contains a lucid account of this work.

Reviewed by Paul Holmes-Higgin, AI Group, Department of Mathematical and Computing Sciences, University of Surrey, UK

Integrated distributed intelligent systems in manufacturing by M. Rao, Q. Wang and J. Cha, Chapman & Hall, London, 1993, pp 324, £35.00, ISBN 0-412-54370-2.

The book first gives a convincing introduction to manufacturing history, what is involved in CIM (Computer-Integrated Manufacturing), the need for AI in manufacturing and issues in Integrated Distributed Intelligent Systems (IDIS) in manufacturing. The authors pay particular attention to