Lecture Notes in Artificial Intelligence

Subseries of Lecture Notes in Computer Science
Edited by J. Siekmann

Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

Editorial

Artificial Intelligence has become a major discipline under the roof of Computer Science. This is also reflected by a growing number of titles devoted to this fast developing field to be published in our Lecture Notes in Computer Science. To make these volumes immediately visible we have decided to distinguish them by a special cover as Lecture Notes in Artificial Intelligence, constituting a subseries of the Lecture Notes in Computer Science. This subseries is edited by an Editorial Board of experts from all areas of AI, chaired by Jörg Siekmann, who are looking forward to consider further AI monographs and proceedings of high scientific quality for publication.

We hope that the constitution of this subseries will be well accepted by the audience of the Lecture Notes in Computer Science, and we feel confident that the subseries will be recognized as an outstanding opportunity for publication by authors and editors of the AI community.

Editors and publisher

Lecture Notes in Artificial Intelligence

Subseries of Lecture Notes in Computer Science
Edited by J. Siekmann

406

C.J. Barter M.J. Brooks (Eds.)

AI '88

2nd Australian Joint Artificial Intelligence Conference
Adelaide, Australia, November 15–18, 1988
Proceedings

Springer-Verlag
Berlin Heidelberg New York London Paris Tokyo Hong Kong

Editors

Christopher J. Barter
Department of Computer Science, University of Adelaide
GPO Box 498, Adelaide, South Australia 5001

Michael J. Brooks
Discipline of Computer Science, Flinders University
Bedford Park, South Australia 5042

CR Subject Classification (1987): I.2

ISBN 3-540-52062-7 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-52062-7 Springer-Verlag New York Berlin Heidelberg

Printing and binding: Druckhaus Beltz, Hemsbach/Bergstr.
2145/3140-543210 – Printed on acid-free paper

Preface

AI '88 is the second Australian Joint Artificial Intelligence Conference to be held. Its broad objective is to bring business, industry and researchers together to consider the current activities and future potential of artificial intelligence, encompassing both practical and theoretical issues.

Many papers were submitted to the conference, including some from as far afield as Canada, France, U.K., U.S.A., Sweden, Italy and Thailand. Each paper was reviewed by at least two independent referees, and approximately a third of those submitted were accepted. We are very grateful to the many referees who assisted and especially to the overseas contingent, and to those that provided urgent or multiple reviews. A list of the AI '88 referees is given overleaf.

AI '88 exhibits papers that cover a wide range of topics, including vision, robotics, knowledge acquisition, expert systems, natural language and reasoning. Contributions vary from the highly theoretical to the most practical of applications.

Many colleagues contributed generously to the organisation of the conference. Particularly valued was the work on the Programme Committee of Dr. S. Hood and Dr. T. Li, and the secretarial assistance of Ms. T. Young.

Chris Barter
Mike Brooks

Contents

Invited Presentations

Expert Systems

Tutoring

Logic and Reasoning

Knowledge Acquisition and Representation

Knowledge Engineering

Vision and Robotics

Natural Language

Parallel Processing and Distributed AI

Invited Presentations

A PERSPECTIVE ON THE NATURE OF ARTIFICIAL INTELLIGENCE
--Enabling and Enhancing Capabilities for Society

William J. Clancey
Institute for Research on Learning
3333 Coyote Hill Road
Palo Alto, CA 94304

Keynote Abstract

Knowledge engineering and more generally artificial intelligence have received considerable attention in the past five years, prompting a burgeoning literature from academia, industrial research labs, and the popular press. Yet for all this, the common understanding about "expert systems" and artificial intelligence research in general is woefully superficial and even misdirected. So much attention has been given to the *goals* of "making intelligent machines" that the actual accomplishments of the field and how its *methods* are different from traditional programming are poorly articulated. Indeed, the important lessons are even prone to being missed entirely by the managers, programmers, scientists, and engineers who are not computer science specialists and most need to understand what is new. In large part, AI researchers are so caught up in their individual projects, they neglect to relate their work to what has gone before.

What do we know about Artificial Intelligence today? A surprising perspective has emerged, with ramifications that are in many respects more important than just producing "intelligent robots," as ironic as that may seem. The new perspective is visible in a few trends. First is the growing subfield of AI called "qualitative reasoning," which emphasizes reasoning about the spatial and temporal relations of causal processes. Strikingly, such programs do not *reason about* systems (such as electronic circuits) so much as *simulate* the physical processes. This is curious because "intelligence" in the field of AI has so often been associated with solving problems, for example diagnosing or designing a circuit. The example is particularly clear when we consider "intelligent tutoring systems" such as SOPHIE or STEAMER, which undeniably use AI programming methods, but have no independent capability to diagnose or control (respectively) the systems they are about. Furthermore, the original SCHOLAR program carries on instructional dialogues about geography and demographics, but has no "problem-solving" capability or internal representations of physical, causal processes. Thus, we have programs that can solve problems, that model physical systems, or just converse about facts.

So what is the unifying concept behind AI research? A straightforward synthesis is possible. AI programs have in common the use of a *qualitative modeling methodology*. That is, we can understand artificial intelligence programming methods as a means for describing processes in a primarily non-numeric (hence, *qualitative*) way. The processes might be physical (such as the steam propulsion plant in STEAMER) or cognitive (such as a diagnostic inference procedure), or they might concern social interactions (such as natural language discourse). Other processes involving perception, motion, or plans for interacting groups of agents have also been modeled using qualitative methods.

Understanding this perspective requires giving up a few overly restricted and confusing points of view which are common in the literature. These changes in perspective include realizing that: 1) all knowledge bases contain models of some system being reasoned about; 2) a model doesn't have to be a simulation; classification models are the most common and for open systems are irreducible to simulations; 3) AI representations can be described as relational networks, of which hierarchies and state-transition networks are the most basic structures; 4) unless we are doing psychology, we are modeling some system in the world, not a domain expert.

Specifically, knowledge engineering is a methodology for acquiring, representing, and using qualitative models of systems. We can understand knowledge engineering as a method for modeling systems in the world by using human experts as informants about 1) familiar situations that occur in the operation of systems and 2) familiar plans for taking actions to construct, control, diagnose, etc. systems. We distinguish between types of systems being modeled (physical, cognitive, social, etc.), modeling tasks (such as diagnosis and control), computational methods (such as heuristic classification), and implementation languages (such as rules and frames).

Besides viewing knowledge bases as qualitative models, a second trend shaping our understanding about the nature of AI is the improved philosophical analysis of representations and programs. The work of Winograd, Brian Smith, and others is forcing many researchers to acknowledge that what domain experts know is both less principled and of a different nature than what we are formalizing in our programs. We are in fact wedged between two untenable analyses. If we insist that programs represent concepts completely, as definitions with necessary and sufficient properties, then concepts in programs are not what human concepts are like, excepting (perhaps) mathematical, axiomatic domains. Human concepts are open, ill-defined, in many respects inseparable from the context of activity in which they are used. The alternative perspective is no better. We can adopt the view that we haven't captured the semantics of concepts in our programs, that tokens like CULTURE and PATIENT are interpretable only by people. Computer programs don't understand the words they use. Knowledge for a person is much more than what we can represent.

This analysis gets subtle very fast and there are good reasons to keep an open mind. Curiously though, this second trend has an important implication for the qualitative modeling perspective, amplifying it and reaffirming what the AI business is about. A perfectly consistent resolution of the philosophical analysis on the nature of AI is to acknowledge that AI programs are just models after all, no more real than a body of numeric equations. Thereby we emphasize that we, as scientists and engineerings, are merely using AI programming techniques as a methodology for going about our business of modeling the world and designing, controlling, diagnosing, or repairing complex systems.

This is something to be very excited about. We are saying that qualitative modeling on a computer is the most radical improvement to our scientific and engineering tools since Newton invented the differential calculus. It's just in time, too. Both the social, economic world and the designed, artifactual world of machines have become too complex for numeric measures to usefully describe. Our predictive, diagnostic and control techniques are being overwhelmed by the sheer combinatoric volume of interactions. Qualitative modeling promises a way out, a way of partially automating these processes, which could radically enhance and augment human capabilities when used to its best advantage.

It's easy to think of *enhancing capabilities*, what qualitative modeling of processes will enable us to do more efficiently, with less waste. For example, in manufacturing, programs can help simulate assembly and estimate repair costs during the design stage, and they can control processes and detect failures earlier and more reliably than people can.

But more interesting are the *enabling capabilities*, what qualitative modeling of processes will enable us to do that otherwise wouldn't have been possible. Consider for example the new product Philips is planning to release this year, called Compact Disc-Interactive (CD-I). Sold as a "player" to consumers, this is actually a workstation for the educational and home entertainment environment. But without far more sophisticated means of indexing and selectively retrieving material (sound, pictures, graphics, movies, etc.) the actual uses for this machine will pale in the face of the huge storage and computational power it will make available to the layman. We have to make much more progress in representing relations among concepts, modeling people's interests and capabilities, and modeling interactional processes by which people access and learn from books, films, and libraries in general.

Another enabling capability involves activity in dangerous environments. Without taking a political stand on nuclear power or manned space exploration, we all have to agree that robots can play a major role in controlling equipment where human life is at risk. Similarly, AI is making possible remote control and sensing of experiments in what is called *telescience*.

The huge complexity of biological and electronic systems makes them natural targets for qualitative modeling. Scientists and engineers are already busy using AI methods to simulate processes and describe incredibly large structures. Examples are the DENDRAL, PROTEAN, MOLGEN, and VLSI projects at Stanford. Not the least important is the merging of numeric and qualitative techniques by which numeric simulations are set up, run, and evaluated under the control of an AI program. For certain appliations, such as product design, traditional linear optimization routines can be usefully automated, allowing the search space of alternative designs to be more exhaustively and rapidly generated and tested. Most people (including the researchers!) often forget that the core of DENDRAL was a molecular structure generator based on group theory and proven to be complete. Feigenbaum may emphasize the power of DENDRAL's heuristics, but it's the mathematically complete model that ultimately makes the program better than people.

Programs themselves have become too complex to manage without automated aids. The graphic aids of workstations for browsing code and knowledge bases will get better as we adopt relational and process-oriented representations within the interface programs themselves. The implications for libraries are far from being realized, though the need and potential for automated browsers is obvious.

Finally, after this somewhat unbridled optimism we must return to the philosophers' lament. If we work at it, computers will enable individual people to deal with the world on a more individual basis, to cope with complexity and raise their attention to higher levels. An expert system at GM, for example, helps mechanics use and integrate the results of complex test equipment. If we design these tools right, they will give people a better grasp of their environment and place them on a level to cope with and understand increasingly more complex problems. But to do this, we must make computers as directly manipulable as hammers and as transparent as the Macintosh. Otherwise, we will alienate man all the more from machine, interposing black boxes that take over any responsbility and prevent whatever meager understanding the poorly trained technician may have been capable of.

We must remember the modeling perspective. The combination of computers' complexity and our generally superficial relationships in society, will all too often lead us to attribute human characteristics to these machines. As researchers, we need to stop reveling in the wonderful effects we can achieve and start adopting the traditional scientific and engineering perspective. For example, we shouldn't accept an AI paper for publication unless the model is pushed until it breaks and the author analyzes why. We need to build in multiple representations so programs can know their boundaries. Computer models should print out a commentary of how good a solution is, where it might be wrong, and what additional knowledge would improve it. No science or engineering could advance without methods for testing a model, and knowledge engineering needs this all the more. Imagine building a bridge that could be moved around the world, placed under different loads, and even restructured for new environments. What kind of a theory of bridges would that require? But this is precisely the kind of robustness we currently expect of an expert system.

We need to stop ignoring the theory of representation as an annoying thorn, and place it in our curriculums with the same importance we give to graph theory and the theory of computation. We won't get our foundations right until we realize that as computational modelers we are by our very nature *formalists* and had better soon adopt a common notation and qualitative calculus. Sowa's conceptual networks are a fine start. More graph and set theory wouldn't hurt.

It's trite to say that we can't ignore learning if we're interested in intelligence. The need will soon be obvious. Once the expert systems are in place it will be clear that they could be better if they analyzed the problems they have solved for patterns and were tuned thereby. It seems plausible that programmed learning from experience will be as commonplace in expert systems as backward chaining before the next decade is out. Recent advances in explanation-based learning suggest a productive merger of classical learning, case-based reasoning, and knowledge acquisition.

So AI programs may be considered to be intelligent for all practical purposes and even be useful assistants. We needn't lament to have lost the superhuman intelligent robot, of which so many AI researchers dreamed. We have gained instead an incredible modeling tool with enhancing and enabling capabilities for society that we hardly imagined of a few decades ago. And maybe in the process, through the