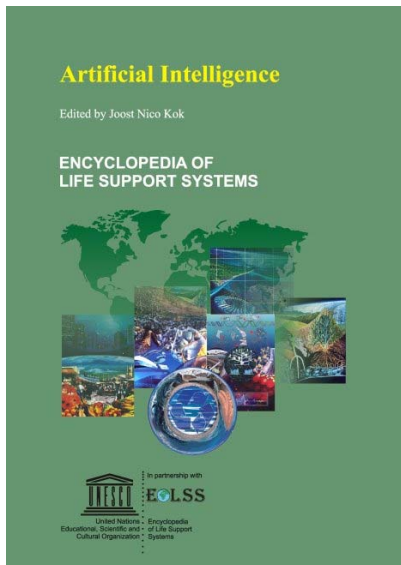


CONTENTS

ARTIFICIAL INTELLIGENCE



Artificial Intelligence - Volume 1

No. of Pages: 401

ISBN: 978-1-84826-125-9 (eBook)

ISBN: 978-1-84826-575-2 (Print Volume)

[For more information of e-book and Print Volume\(s\) order, please click here](#)

[Or contact : eolessunesco@gmail.com](mailto:eolessunesco@gmail.com)

CONTENTS

Artificial Intelligence: Definition, Trends, Techniques, and Cases	1
Joost N. Kok, <i>Leiden Institute of Advanced Computer Science, Leiden University, the Netherlands</i>	
Egbert J. W. Boers, <i>Leiden Institute of Advanced Computer Science, Leiden University, the Netherlands</i>	
Walter A. Kusters, <i>Leiden Institute of Advanced Computer Science, Leiden University, the Netherlands</i>	
Peter van der Putten, <i>Leiden Institute of Advanced Computer Science, Leiden University, the Netherlands</i>	
Mannes Poel, <i>Faculty of Computer Science, University of Twente, the Netherlands</i>	

1. Introduction
 - 1.1. Definition of Artificial Intelligence
 - 1.2. The Turing Test
 - 1.3. Final Introductory Remarks
2. A Short History of Artificial Intelligence
3. Current Trends in Artificial Intelligence
4. Techniques
 - 4.1. Natural Computation
5. Cases
 - 5.1. Checkers
 - 5.2. Expert Systems and Belief Networks
 - 5.3. A Virtual Company
 - 5.4. RoboCup
6. Concluding Remarks

Logic in AI	21
J. J. Ch. Meyer, <i>Department of Computing Science, Utrecht University, The Netherlands</i>	

1. Introduction
2. Classical Logic
 - 2.1. Classical propositional logic
 - 2.2. First-Order Predicate Logic
3. Modal Logic
 - 3.1. Dynamic logic
 - 3.2. Deontic logic
 - 3.3. Temporal logic
 - 3.4. Epistemic logic
4. Nonmonotonic Logic
 - 4.1. Default Logic
 - 4.2. Circumscription
5. Multi-Valued Logic and Fuzzy Logic
6. Conclusion

Intelligent Agents	44
Catholijn M. Jonker, <i>Vrije Universiteit Amsterdam, Department of Artificial Intelligence, Amsterdam, The Netherlands</i>	
Jan Treur, <i>Vrije Universiteit Amsterdam, Department of Artificial Intelligence, Amsterdam, The Netherlands</i>	

1. Introduction
2. Agent Notions
 - 2.1. Weak Notion of Agent
 - 2.2. Other Notions of Agent
3. Primitive Agent Concepts
 - 3.1. External primitive concepts
 - 3.2. Internal primitive concepts

- 3.3. An Example Analysis
 - 3.3.1. The elevator in terms of external primitive concepts
 - 3.3.2. The elevator in terms of internal primitive concepts
 - 3.3.3. Types of behaviour of the elevator
- 4. Business Websites
- 5. A Generic Multi-Agent Architecture for Intelligent Websites
- 6. Requirements for the Website Agents
 - 6.1. Characteristics and Requirements for the Website Agents
 - 6.2. Characteristics and Requirements for the Personal Assistants
- 7. The Internal Design of the Information Broker Agents
 - 7.1. A Generic Information Broker Agent Architecture
 - 7.2. The Website Agent: Internal Design
 - 7.3. The Personal Assistant: Internal Design

Dynamical Systems, Individual-Based Modeling, and Self-Organization

64

Edwin D. de Jong, *Artificial Intelligence Laboratory, Vrije Universiteit Brussel, Belgium*
 Bart G. de Boer, *Artificial Intelligence Laboratory, Vrije Universiteit Brussel, Belgium*

- 1. Introduction
- 2. Individual-based modeling
- 3. Basic notions of dynamical systems theory
 - 3.1. Dynamical system
 - 3.2. State variable
 - 3.3. Dimension
 - 3.4. Control parameter
 - 3.5. Control law
 - 3.6. Determinism
 - 3.7. Non-determinism
 - 3.8. Differential equation
 - 3.9. Difference equation
 - 3.10. Autonomous dynamical systems
 - 3.10.1. Pendulum
 - 3.11. Linear dynamical systems
 - 3.12. Phase plot
 - 3.13. State space (Phase space)
 - 3.14. Trajectories
 - 3.14.1. Phase plots of the pendulum
 - 3.15. Dissipative systems
 - 3.16. Bifurcation
- 4. Non-linear dynamical systems
 - 4.1. Phase portrait
 - 4.2. Fixed points
 - 4.3. Stability
 - 4.4. Point attractor
 - 4.5. Repellor
 - 4.6. Saddle point
 - 4.7. Basin of attraction
 - 4.7.1. Phase portrait of a dynamical system
 - 4.7.2. Other forms of attractors
 - 4.8. Limit cycle
- 5. Other attractors
 - 5.1. Strange attractor
 - 5.2. Fractal set
 - 5.3. Chaotic attractor
 - 5.4. Lyapunov exponent
- 6. Chaos
 - 6.1. The Lorenz attractor

- 6.2. The logistic map
- 7. Self-organization
 - 7.1. Self-organization (emergence)
 - 7.1.1. Coherent behavior (Synergetics)
 - 7.1.2. Dissipative structures
 - 7.1.3. Entropy
 - 7.2. Complexity
 - 7.3. The edge of chaos (Complexity)
 - 7.4. Self-organized criticality
 - 7.4.1. Catastrophe
 - 7.4.2. Zipf's law
- 8. Examples of self-organization and chaos
 - 8.1. The Belousov-Zhabotinski reaction
 - 8.2. Rayleigh-Bénard convection
 - 8.3. Turing patterns
 - 8.4. Self-organization in the brain
 - 8.5. The honeycomb
 - 8.6. Traffic
- 9. A Software simulation environment: Swarm

Machine Learning**89**

Lorenza Saitta, *Universit del Piemonte Orientale, Dipartimento di Scienze e Tecnologie Avanzate, Alessandria, Italy*

- 1. Introduction
- 2. Basic Knowledge Representation and Learning Methods
 - 2.1. Concept Learning
 - 2.2. Concept Formation
 - 2.3. Evolutionary Learning
 - 2.4. Reinforcement Learning
 - 2.5. Case-Based Learning
 - 2.6. Neural Networks
- 3. Current Focus of Research
- 4. Major Achievements and Current Trends

Computational Intelligence**105**

B.C. Craenen, *Faculty of Exact Sciences, Vrije Universiteit Amsterdam, The Netherlands*
 A.E. Eiben, *Faculty of Exact Sciences, Vrije Universiteit Amsterdam, The Netherlands*

- 1. Introduction
- 2. What is computational intelligence?
- 3. Artificial versus computational intelligence
- 4. Computational intelligence subfields from different perspectives
- 5. Activities in computational intelligence
- 6. Concluding remarks

Evolutionary Computation**118**

Th. Back, *Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands*

- 1. A General Evolutionary Algorithm
- 2. Classical Genetic Algorithms
 - 2.1. The Structure of Individuals
 - 2.2. Mutation
 - 2.3. Recombination
 - 2.4. Selection

3. Evolution Strategies
 - 3.1. The Structure of Individuals
 - 3.2. Mutation
 - 3.3. Recombination
 - 3.4. Selection
4. Evolutionary Programming
 - 4.1. The Structure of Individuals
 - 4.2. Mutation
 - 4.3. Recombination
 - 4.4. Selection
5. Genetic Programming
 - 5.1. The Structure of Individuals
 - 5.2. Mutation
 - 5.3. Recombination
 - 5.4. Selection
6. Theory of Evolutionary Algorithms
 - 6.1. Convergence Velocity
 - 6.2. Convergence Velocity Analysis of Genetic Algorithms
 - 6.3. Classical Genetic Algorithm Theory
 - 6.4. Global Convergence Properties
7. Applications
8. Summary

Quantum Computing

155

Paul Vitanyi, *CWI, Kruislaan, Amsterdam, The Netherlands*

1. Introduction
 - 1.1. Wither Classical Computing?
 - 1.1.1. The Cooking Problem
 - 1.1.2. The Spaghetti Problem
2. Quantum Computation
 - 2.1. Background: Probabilistic Turing Machines
 - 2.2. Quantum Turing Machines
 - 2.3. Multi-Qubit states
 - 2.4. Input and Output
 - 2.5. Quantum Algorithmics
3. What quantum computers can do
 - 3.1. Fast Factoring
 - 3.2. Cryptography
 - 3.3. Rapid Search
 - 3.4. Communication Complexity
 - 3.5. Simulating Quantum Matter
 - 3.6. Limitations
4. How Quantum Computers Do It

Neural Networks

173

Tom Heskes, *SNN, University of Nijmegen, The Netherlands*
 David Barber, *Aston University, Birmingham, UK*

1. What are neural networks?
2. Neurobiology
 - 2.1. Neurons
 - 2.2. Simple Neuron Models
3. The Hopfield Network
 - 3.1. The Hebb rule
4. Statistical Physics

- 4.1. The Boltzmann Machine
- 4.2. Statistical Mechanics of Ising Systems
- 4.3. Markov Random Fields
- 5. Perceptrons
 - 5.1. The Perceptron
 - 5.2. Multilayer Perceptrons
- 6. Training Neural Networks
 - 6.1. Neural Networks as Advanced Statistical Tools
 - 6.1.1. Training multi-layered perceptrons
 - 6.1.2. Statistical inference and overfitting
 - 6.2. The Classical Approach
 - 6.2.1. Regularisation
 - 6.2.2. Model choice
 - 6.2.3. Training ensembles of networks
 - 6.3. The Bayesian Framework
 - 6.3.1. The Bayesian philosophy
 - 6.3.2. Prior and posterior
 - 6.3.3. Bayesian model selection
 - 6.3.4. An example
 - 6.3.5. Sampling Methods
- 7. Neural Network Applications
 - 7.1. General considerations
 - 7.2. Example applications
- 8. Summary and Outlook

Fuzzy Logic**211**

Michael R. Berthold, *Berkeley Initiative in Soft Computing (BISC), University of California at Berkeley, USA*

- 1. Introduction
- 2. Basics of Fuzzy Sets and Fuzzy Logic
- 3. Linguistic Variables and Fuzzy Sets
- 4. Fuzzy Numbers
- 5. Fuzzy Sets and Fuzzy Logic
- 6. Generalized Modus Ponens
- 7. Fuzzy Rules
- 8. Fuzzy Graphs
- 9. Fuzzy Clustering
- 10. Fuzzy Decision Trees

DNA Computing**230**

H. J. Hoogeboom, *Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands*
 G. Rozenberg, *Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands*

- 1. Introduction
- 2. DNA molecules and their processing
 - 2.1. Structure of DNA molecules
 - 2.2. Notation for DNA molecules
 - 2.3. Processing of DNA molecules
- 3. Computing with DNA molecules
 - 3.1. Graphs
 - 3.2. DNA based Algorithm for the HPP
- 4. Bottleneck of traditional computing
- 5. Discussion

Knowledge Based System Development Tools

247

John K.C. Kingston, *AIAI, University of Edinburgh, Scotland*

1. Introduction
2. KBS Tools: Functionality
 - 2.1. Production rules; forward and backward chaining
 - 2.2. Object oriented programming
 - 2.3. Hypothetical reasoning
3. KBS Tools: Classification
 - 3.1. Classifying KBS tools: Shells
 - 3.2. Classifying KBS Tools: Procedural Languages
 - 3.3. Classifying KBS Tools: Toolkits
 - 3.3.1. Classifying KBS Tools: AI Languages
 - 3.4. Classifying KBS Tools: Specialised tools
 - 3.4.1. Domain-specific and task-specific tools
 - 3.4.2. Paradigm-specific tools
 - 3.4.3. Internet based tools
 - 3.5. Classifying KBS tools: ART-like and KEE-like
 - 3.5.1. ART Camp
 - 3.5.2. KEE Camp
4. Selecting a KBS tool
 - 4.1. Selecting KBS: Features of KBS tools
 - 4.1.1. Flexibility
 - 4.1.2. Ease of Use
 - 4.1.3. Efficiency
 - 4.1.4. Extendability
 - 4.1.5. Support
 - 4.1.6. Portability
 - 4.1.7. Reliability
5. Selecting KBS Tools
 - 5.1. Selecting KBS: Features of the Problem
 - 5.1.1. Task Type
 - 5.1.2. Other features of the problem
 - 5.2. Selecting KBS: Phase of Development
 - 5.2.1. Exploration
 - 5.2.2. Prototyping
 - 5.2.3. Development
 - 5.2.4. Fielding
 - 5.2.5. Operation
 - 5.2.6. Relationship between Development Phases and Tool Features
 - 5.3. Selecting KBS: Organisational policies and capabilities
 - 5.3.1. Cost
 - 5.3.2. Development Team
6. Conclusion

Speech Processing

274

J. R. de Pijper, *IPO, Eindhoven, The Netherlands*E. Klabbers, *Center for Spoken Language Understanding, OGI School of Science and Engineering, Beaverton, USA*

1. Introduction
 - 1.1. Speech in the user interface
2. Speech synthesis
 - 2.1. Introduction
 - 2.2. Diphone concatenation
 - 2.3. Phrase concatenation
 - 2.4. Unit concatenation

- 2.5. Conclusion
- 3. Speech recognition
 - 3.1. Introduction
 - 3.2. How does it work
 - 3.3. Performance of ASR
- 4. Application areas
 - 4.1. Dictation systems
 - 4.2. Text-to-Speech
 - 4.3. Voice-command systems
 - 4.4. Dialogue systems
 - 4.5. Speaker identification and verification
- 5. A complete dialogue system: OVIS

Data Mining**287**

Elena Marchiori, *Free University Amsterdam, Faculty of Science, Department of Mathematics and Computer Science, Amsterdam, The Netherlands*

- 1. Introduction
- 2. Goals
- 3. Techniques
 - 3.1. Query Tools
 - 3.2. Visualization Techniques
 - 3.3. K-Nearest Neighbor
 - 3.4. Decision Trees
 - 3.5. Association Rules
 - 3.6. Inductive Logic Programming
 - 3.6.1. Inverse Resolution
 - 3.6.2. Top Down Learning
 - 3.7. Neural Networks
 - 3.8. Genetic Algorithms
- 4. Applications
- 5. Conclusion

Vision**300**

Bart M. ter Haar Romeny, *Faculty of Biomedical Engineering, Eindhoven Technical University, the Netherlands*

- 1. Introduction
- 2. The eye
- 3. The retina
- 4. Retinal receptive fields
- 5. The Lateral Geniculate Nucleus
- 6. The primary visual cortex
- 7. Intermezzo: Measurement of neuronal activity
- 8. Hypercolumns and orientation structure
- 9. Higher cortical visual areas
- 10. Computer Vision - Computational Vision
- 11. Non-linear, geometry-driven diffusion
- 12. Multiscale measurement and deep structure
- 13. Outlook

Expert Systems**328**

Peter Lucas, *Institute for Computing and Information Sciences, Radboud University Nijmegen, the Netherlands*

1. Introduction
 - 1.1. Definition of the Field
 - 1.2. Origin and Evolution
2. Expert System Principles
 - 2.1. Expert System Architecture
 - 2.2. Problem-solving Methods
3. Knowledge Representation and Inference
 - 3.1. Logic Representation and Reasoning
 - 3.1.1. Horn-clause Logic
 - 3.1.2. Objects, Attributes and Values
 - 3.2. Diagnostic Problem Solving
 - 3.2.1. Deductive Diagnosis
 - 3.2.2. Abductive Diagnosis
 - 3.2.3. Consistency-based Diagnosis
 - 3.3. Configuring and Design
 - 3.4. Uncertainty
 - 3.5. Decision Making
4. Knowledge Engineering
5. Conclusions

Index 357

About EOLSS 365