

Abstracts of all articles for Cognitive Systems 2

Denise Peters, Sven Bertel, Thomas Barkowsky, Christian Freksa

1 Introduction to Soar

A Gentle Introduction to SOAR, an Architecture for Human Cognition: 2006 Update This article provides an introduction to the ideas of cognitive modeling with cognitive architectures in general and with the SOAR architecture (Version 9) in particular. A number of general key properties of cognitive behaviours are identified and mapped onto actions that are carried out in specific problem spaces. Various aspects of memory, perception, action and cognition are discussed.

2 SOAR Tutorial part I and II:

Introduction This is a guide for learning to create software agents in Soar, version 8. It assumes no prior knowledge of Soar or computer programming. What is Soar? We call Soar a unified architecture for developing intelligent systems. That is, Soar provides the fixed computational structures in which knowledge can be encoded and used to produce action in pursuit of goals. In many ways, it is like a programming language, albeit a specialized one. It differs from other programming languages in that it has embedded in it a specific theory of the appropriate primitives underlying reasoning, learning, planning, and other capabilities that we hypothesize are necessary for intelligent behavior. Soar is not an attempt to create a general purpose programming language. You will undoubtedly discover that some computations are difficult or awkward to do in Soar (such as complex math) and they are more appropriately encoded in a programming language such as C, C++, or Java. Our hypothesis is that Soar is appropriate for building autonomous agents that use large bodies of knowledge to generate action in pursuit of goals.

Eaters World The eater world consists of a rectangular grid, 15 squares wide by 15 squares high. Walls bound all four sides. Interior wall sections are randomly generated for each new game. No two walls will touch, so there are no corners, except for exterior walls and no dead ends anywhere on the board. Each eater starts at a random location. Food pellets are in all other squares of the grid. There are two kinds of food: normal food (blue circles and worth 5 points) and bonus food (red stars and worth 10 points). An eater consumes food by moving into a square. When an eater moves out of a square it will be empty (unless another eater moves into it).

3 Introduction to ACT-R:

An Integrated Theory of the Mind Adaptive control of thought/rational has evolved into a theory that consists of multiple modules but also explains how these modules are integrated to

produce coherent cognition. The perceptual-motor modules, the goal module, and the declarative memory module are presented as examples of specialized systems in ACTR. These modules are associated with distinct cortical regions. These modules place chunks in buffers where they can be detected by a production system that responds to patterns of information in the buffers. At any point in time, a single production rule is selected to respond to the current pattern. Subsymbolic processes serve to guide the selection of rules to fire as well as the internal operations of some modules. Much of learning involves tuning of these subsymbolic processes. A number of simple and complex empirical examples are described to illustrate how these modules function singly and in concert.

4 ACT-R Tutorial I and II:

Production Systems ACT-R is a production system theory that tries to explain human cognition by developing a model of the knowledge structures that underlie cognition. There are two types of knowledge representation in ACT-R – declarative and procedural knowledge. Declarative knowledge corresponds to things we are aware we know and can usually describe to others. Examples of declarative knowledge include George Washington was the first president and An atom is like the solar system. Procedural knowledge is knowledge which we display in our behavior but which we are not conscious of. For instance, no one can describe the rules by which we speak a language and yet we do. In ACT-R declarative knowledge is represented in structures called chunks whereas procedural knowledge is represented in productions. Thus chunks and productions are the basic building blocks for an ACT-R model.

Perception and Motor Actions in ACT-R This unit will discuss how ACT-R can interact with experimental software. This is made possible with the addition of ACT-R/PM (the PM stands for perceptual-motor), which was developed by Mike Byrne. It is a set of modules, which act in parallel with the ACT-R production system for controlling vision, motor, audition, and speech. It also contains the code necessary to allow those modules to interact with the computer i.e. see what is on the screen, press keys, and move and click the mouse. It is quite elaborate and we will not be describing the entire system in the tutorial. For more complete information you can check out the ACT-R/PM web site at <http://chil.rice.edu/byrne/RPM/index.html>.

5 ACT-R Tutorial III and IVold:

Attention This unit is concerned with developing a better understanding of how perceptual attention works in ACT-R, particularly as it is concerned with visual attention.

Complex Processing This unit is concerned with how we engage in a complex stream of processing. We will discuss how ACT-R can read sentences, extract their meaning, and later retrieve these meanings. We will also cover the buffer stuffing mechanism as it relates to the visual system.

6 ACT-R Tutorial IV and V:

Activation of Chunks and Base-Level Learning There are two goals of this unit. The first is to introduce the subsymbolic quantity of activation associated with chunks. The other is to show how those activation values are learned through the history of usage of the chunks.

Activation and Context The goal of this unit is to introduce the components of the activation equation that reflect the context of a declarative memory retrieval.

7 An ACT-R Application :

Student Modeling in the ACT Programming Tutor The goal of this project is to model a student's changing knowledge state during cognitive skill acquisition. Our purpose is to predict student's performance levels and to implement a mastery-based learning environment that enables students to achieve satisfactory performance levels in a minimum period of time. The project assumes a production system model of skill knowledge in which if-then production rules associate problem states and goals with actions and consequences. Our diagnostic goal is to model the student's knowledge of these rules and to model changes in the student's knowledge state over time. The project focuses on students learning to write programs in LISP with assistance of an intelligent computer-based programming tutor. In the following section we describe the learning environment, the curriculum, the cognitive model and the learning and performance assumptions. Finally we discuss our evaluations of the model.

8 Spatial Representation in ACT-R

Mechanisms for Human Spatial Competence Research spanning decades has generated a long list of phenomena associated with human spatial information processing. Additionally, a number of theories have been proposed about the representation, organization and processing of spatial information by humans. This paper presents a broad account of human spatial competence, integrated with the ACT-R cognitive architecture. Using a cognitive architecture grounds the research in a validated theory of human cognition, enhancing the plausibility of the overall account. This work posits a close link of aspects of spatial information processing to vision and motor planning, and integrates theoretical perspectives that have been proposed over the history of research in this area. In addition, the account is supported by evidence from neuropsychological investigations of human spatial ability. The mechanisms provide a means of accounting for a broad range of phenomena described in the experimental literature.

9 Introduction to connectionist models

Artificial neural networks: fundamentals, computing, design, and application Artificial neural networks (ANNs) are relatively new computational tools that have found extensive utilization in solving many complex real-world problems. The attractiveness of ANNs comes from their remarkable information processing characteristics pertinent mainly to nonlinearity, high parallelism, fault and noise tolerance, and learning and generalization capabilities. This paper aims to familiarize the reader with ANN-based computing (neurocomputing) and to serve as a useful companion practical guide and toolkit for the ANNs modeler along the course of ANN project development. The history of the evolution of neurocomputing and its relation to the field of neurobiology is briefly discussed. ANNs are compared to both expert systems and statistical regression and their advantages and limitations are outlined. A birds eye review of the various types of ANNs and the related learning rules is presented, with special emphasis on backpropagation (BP) ANNs theory and design. A generalized methodology for developing successful ANNs projects from conceptualization, to design, to implementation, is described. The most

common problems that BPANNs developers face during training are summarized in conjunction with possible causes and remedies.

10 Introduction to JavaNNS I

A Quick Tour of JavaNNS JavaNNS is a simulator for artificial neural networks, i.e. computational models inspired by biological neural networks. It enables you to use predefined networks or create your own, to train and to analyze them. This tutorial is an introduction to JavaNNS. On the example of a XOR network we will introduce you to all possibilities of the simulation and show you how to analyze a neural network. Also you learn how to build and train your own networks. Finally we introduce backpropagation as one of the important learning rules.

11 The role of neural networks in cognitive science

Networks and Theories: The Place of Connectionism in Cognitive Science What is the proper role of connectionism in cognitive science? This issue is complex and multifaceted, and I will not endeavor to explore all its ramifications (nor am I capable of doing so). Instead, I attempt to develop two specific points. First, I suggest that connectionist networks should not be viewed as theories of human cognitive functions, or as simulations of theories, or viewed as simulations of theories, or even as demonstrations of specific theoretical points. Second, I argue that these networks nevertheless hold considerable promise as tools for development of cognitive theories.

12 Intelligence without representation

Artificial intelligence research has foundered on the issue of representation. When intelligence is approached in an incremental manner, with strict reliance on interfacing to the real world through perception and action, reliance on representation disappears. In this paper we outline our approach to incrementally building complete intelligent Creatures. The fundamental decomposition of the intelligent system is not into independent information processing units which must interface with each other via representations. Instead, the intelligent system is decomposed into independent and parallel activity producers which all interface directly to the world through perception and action, rather than interface to each other particularly much. The notions of central and peripheral systems evaporate everything is both central and peripheral. Based on these principles we have built a very successful series of mobile robots which operate without supervision as Creatures in standard office environments.

13 Doing without Representing?

Connectionism and classicism, it generally appears, have at least this much in common: both place some notion of internal representation at the heart of a scientific study of mind. In recent years, however, a much more radical view has gained increasing popularity. This view calls into question the commitment to internal representation itself. More strikingly still, this new wave of anti-representationalism is rooted not in 'armchair' theorizing but in practical attempts to model and understand intelligent, adaptive behavior. In this paper we first present, and then critically assess, a variety of recent anti-representationalist treatments. We suggest that so far, at least, the sceptical rhetoric outpaces both evidence and argument. Some probable causes of this premature

scepticism are isolated. Nonetheless, the anti-representationalist challenge is shown to be both important and progressive insofar as it forces us to see beyond the bare representational / non-representational dichotomy and to recognize instead a rich continuum of degrees and types of representationality.

14 An Application of a Cognitive Model

Risk Analysis of Ship Navigation by Use of Cognitive Simulation This paper presents a cognitive simulation approach to risk analysis for maritime operations. For this purpose, a ship navigators cognitive model was constructed for simple course-tracking task based on cognitive task analysis of experimental navigation sessions using a maritime simulator. It describes the dynamic interaction between the navigators cognitive processes and the states during manoeuvring. In addition to performance modelling of normal operations, the model is equipped with a navigator-error generating process to simulate identification error of vessel position and heading and look-ahead error of the future track. The cognitive model was examined in terms of its descriptive and predictive abilities of ship motion and navigators behaviour through simulation runs using identical scenarios for the experimental navigation sessions. Many varieties of simulation were then performed with different navigation scenarios changing the navigators individual factors and manoeuvring conditions to identify critical risk factors at sea. Based on these results, we discuss the possibility of applying the cognitive simulation approach to risk analysis in ship navigation.