REPRESENTATIONAL ISSUES IN MACHINE LEARNING

Devika Subramanian[1]
Department of Computer Science
Cornell University
Ithaca, NY 14853

OBJECTIVES

Present day learners depend on careful vocabulary engineering for their success. The aim of this workshop is to study the nature of the contribution representation makes to learning, with a view to automating the process of representation selection and design for autonomous learners. This would clarify representational practice in machine learning by explicating the relationship between inference, both deductive and inductive, and choice of representation.

ISSUES AND APPROACHES

It is an old cliche that vocabulary choice is the principal determinant of success in learning systems. The various inductive algorithms [Winston, Mitchell, Quinlan] derive their power from appropriate choices for instance and hypothesis languages. In recent years this connection has been *analyzed* in qualitative ways as in [Seely Brown & Lenat] who demonstrate that the success of the discovery system AM was due to the syntax of Lisp mirroring the semantics of the domain of discovery (the theory of numbers), and in more quantitative ways by [Valiant, Haussler] who provide a rigorous analysis of the difficulty of learning concepts from examples in terms of the complexity of the concept language.

To build effective learners we also need methods for the *synthesis* of appropriate representations from descriptions of the learning task. Presently, *we* build in representations of the world into our agents, thus their adaptiveness is severely limited by the initial conceptualizations given to them. Working within a fixed conceptual framework causes two problems. In deductive contexts, an inappropriate choice of formulation has adverse effects on the computational efficiency of solution. Solving the missionaries and cannibals problem [Amarel] is much easier if we conceptualize the problem in terms of *sets* of missionaries and cannibals and not as *individuals.* In inductive contexts, an inadequate set of primitive concepts may make a target concept inexpressible and thus unlearnable. The LeX system [Mitchell] could not describe an integration heuristic involving odd powers of the cosine function, because its vocabulary did not include the concept *odd integer.* Additionally, with the right set of primitives, the simplest methods of induction suffice [Quinlan].

There are two main questions in the synthesis and analysis of representations for learners that this workshop addresses. We then present a brief summary of the approaches to these questions taken by the papers in this volume.

- Synthesis: How are effective representations to be synthesized from descriptions of the class of goals to be achieved? Almost all the research in this area examines the issue of *modifying* an existing representation of a problem to obtain better solutions to the goals (this is the area of inductive reformulation), or to achieve the same solutions faster (deductive reformulation). The problem of building appropriate high-level descriptions of the world from sense-data, is an area that has received less attention.

- Analysis: How do we determine the effectiveness of a representation for a certain class of goals? To answer this question, we need tools and frameworks for describing representations and problem solving methods. The chief issues are the methods for detecting the fact that a representation is inadequate for a task, scaling learning algorithms to cover richer representations, and computing bounds on the complexity of a learning task as a function of the representation language used.

SYNTHESIS

- **Deductive Reformulations**

  The question addressed is: find a representation of a problem domain that allows for a given set of goals to solved within some computational constraints. There are three cases: the given representation is too detailed for the task at hand and some form of abstraction needs to be generated, the representation needs to be reformulated into terms that are efficiently recognizable or executable by an agent, or the representation is too abstract and making finer distinctions allows for more efficient computation.

  One approach to the problem of generating abstractions relies on the intuition that inefficient formulations make irrelevant distinctions. Subramanian presents a method of generating abstraction reformulations that detects and removes detail irrelevant to a given task. A related approach generates a series of abstractions of the original representation that introduces detail in a gradual and systematic manner. Riddle decomposes a problem into a series of subproblems each of which is easier to solve than the original, and whose solutions can be composed to yield the final answer. Mohan and Tong provide a method that reformulates a generate and test algorithm into a hierarchy of abstract generate and test algorithms. The work discussed above focuses on generating abstract concept languages. Callan, on the other hand abstracts the language used for describing instances to make a system that learns from examples more efficient.

  Hsu examines the issue of compiling high level advice to a more efficient representation and proposes the use of partial evaluation to generate production systems and combinational circuits from descriptions of a domain in first-order logic.

  Schlimmer and Woodfill introduce methods that add new distinctions to make problem solving more efficient. Schlimmer's system, BUMBLE, uses the problem solving trace to form classes like "the left edge of the board" from a more abstract class like "the edge of the board", to generate moves quickly. Woodfill reformulates a special class of theories in first-order logic by introducing new objects and relations that are not constructible from the original theory. This results in improvements in space and time complexity for a given class of goal queries.

- **Inductive Reformulations**

  One question addressed here is: finding a compact and simple concept representation given a labeled description of the instances of the concept The chief problem is the generation of high-level features from the lower level features that characterize the instances. Rendell provides an excellent overview of this problem (called constructive induction) and presents a framework to compare systems that perform constructive induction. The constraint on the generation of the new features is that they should map the highly disjunctive descriptions of a concept in the feature space of instances, into a simpler (fewer disjuncts) one. Matheus describes a general purpose constructive induction engine and demonstrates it in the context of improving decision trees. Mehra examines the constructive induction problem in the context of a two-class discrimination learning task. The problem is to find new features to separate the two classes by a hyperplane. He inverts the traditional mapping of instances into feature space and uses special properties of the inverted space to generate the right features. Kokar proposes a method of forming new compounds of given representational elements to make an invariance hold. For example, in dimensionless analysis, the new terms formed are dimensionless constants. Raedt describes a form of constructive induction, where the fact that several concepts in a domain share the same structure is exploited. From each solved concept learning problem, a second order schema is learned and applied to future concept learning situations to learn or invent new concepts.

  In the research described above, the goals of a learner are specified by the designer of the learner. Keller examines the formulation of new learning tasks for an autonomous learning agent. He represents the learner's abstract goal of improving the performance of a problem solver and derives specific goals from this, and knowledge of how the problem solver works.

  Another question is the acquisition of new concepts that capture important regularities in sense-data. Sainte-Marie builds new terms from primary observations to create a half-order theory with which to bias further new term generation. His notion of a perceived dependency is related to determinations [Russell]. Tallis' system starts with a set of primitive undefined symbols, called the set of dimensions and learns their meaning by observing interactions between them in the world. An algorithm for learning these determination-like regularities is provided. Russell and Grosof present an algorithm for learning higher-order determinations, an extension of their previous work on acquiring simpler forms of declarative bias [Russell].

ANALYSIS

- **Evaluating Representations**
  Methods of evaluating the effectiveness of representations is critical for the success of systems that generate or modify their vocabulary. Saxena studies the problem of choosing an instance language from a pre-determined set of such languages. He develops criteria based on the simplicity of the resulting concept description that are similar to Rendell's metric of reducing the number of disjuncts. Chrisman examines the question of detecting an overly strong bias in the concept language in the context of pac-learning [Valiant, Haussler] and relaxing it by moving to a more expressive concept language. His theoretical results indicate that a learner can detect that its concept language is too restrictive much before it examines all the instances required to learn the concept. When should an explanation-based learner add a new learned rule into its knowledge base? Greiner analyzes the conditions under which it is useful (with respect to his model of problem solving) to add new rules by estimating the time/space effects over a given distribution of queries.

- **Frameworks for Representations and Problem Solvers**
  Research on the synthesis of new representations relies rather heavily on the foundations of representation and problem solving. Analyzing the effectiveness of a representation in the context of a problem solving situation, requires detailed models of the problem solver itself. What is the space of possible representations for a concept? How can we estimate the complexities of computing a certain class of goals in a representation with a problem solver? These are the questions addressed here. Holte and Zimmer present first steps towards building a theory of representation based on category theory. Mostow describes a set of primitives for describing search algorithms.

- **Extensions of Learning Methods to Richer Representations**
  Standard problems and methods in learning are re-examined with a view to extending their range of applicability. Maclin and Shavlik describe an improvement to explanation-based learning (EBL) that acquires recursive and iterative concepts. They demonstrate their method in the context of an automatic programming system that learns functions from examples. Dietzen and Pfenning extend EBL to learn higher-order and modal concepts. They implement the extension as a meta-interpreter in λ-Prolog. Morik describes a mechanism that acquires the vocabulary for learning a concept along with the actual definition of the concept in that vocabulary. This is an interesting extension of well-known methods of concept formation and would be useful in the construction of autonomous learners. There are two methods of controlling the combinatorics in the problem of generating a concept description from examples: reformulate the concept language (as was done in the section on synthesis) or alternatively, reduce the branching factor of the existing space by careful choice of the generalization operations. This option is explored by Gordon who shows that this method can, in some situations, be as effective as, and cheaper than changing the hypothesis language.

FUTURE DIRECTIONS

The papers presented at the workshop indicate that the study of representational effects in machine learning is flourishing. Avenues for future research include the automatic generation of approximate representations and the reformulation of concept descriptions to make learning by analogy more effective [Russell & Subramanian]. Other frameworks for analyzing representations, as well as more sophisticated models of problem solving need to be developed to facilitate further progress in this field. The analysis of the tradeoffs in expressive power and the computational efficiency of inductive inference is yet another open problem. The art of vocabulary choice and representation design for learning systems will (hopefully!) move closer to being a science.

References

S. Amarel. On the Representation of Problems of Reasoning about Action, In *Machine Intelligence 3*, edited by D. Michie, University of Edinburgh Press, 1968.

D. Haussler. Applying Valiant's framework to AI concept learning problems, Technical report UCSC-CRL-87-11, Computer Science Department, University of California at Santa Cruz, 1987.

T.Mitchell. Version spaces: an approach to concept learning, PhD thesis, Stanford, 1978.

T. Mitchell, P. Utgoff, R. Banerji. Learning by Experimentation: Acquiring and Refining Problem-Solving Heuristics, in Machine Learning, Vol 1, edited by Michalski, Carbonell and Mitchell, Tioga Press, 1983.

J. Quinlan. Classification in Chess End Games, in Machine Learning, Vol 1, edited by Michalski, Carbonell and Mitchell, Tioga Press, 1983.

S.J. Russell and B. Grosof. A declarative approach to bias in concept learning, AAAI-87.

S.J. Russell and D. Subramanian. Mutual constraints on representation and inference, *Proceedings of the Workshop on Machine learning, meta reasoning and logics*, Portugal, 1988.

J. Seely Brown & D. Lenat. Why AM and Eurisko appear to work, *Artificial Intelligence*, Vol 23, 1984.

P.E. Utgoff. Shift of Bias in Inductive Learning, PhD thesis, Rutgers University, CBM-TR-145, 1983.

L. Valiant. A theory of the learnable, In *Communications of the ACM*, Vol 27, No. 11, 1984