Introduction: Special Issue on Computational Learning Theory

This special issue of *Machine Learning* provides a small sampling of some of the current research in computational learning theory. The papers were selected from among those that appeared in preliminary form in the Proceedings of the 1989 Workshop on Computational Learning Theory (COLT '89), and were presented at the workshop in Santa Cruz, CA, in August of 1989.

The annual COLT workshops are rather broad in scope, and encompass a wide variety of learning models and protocols, learning tasks, and techniques for the analysis of learning algorithms. Perhaps the only common trait that the papers from COLT '89 share is that they all contain formal analyses of theoretical issues in machine learning. Most (although not all) of the COLT papers focus on the general problem of *concept learning*, and it is this area that has been evolving most rapidly in the last few years. Much of the reason for this progress is the development of various formalizations of the concept learning problem that capture many of the characteristics of what concept learning means intuitively, while at the same time allow for successful application of rigorous mathematical techniques to a variety of specific learning problems. A main goal of computational learning theory is the suitable formalization of other important aspects of machine learning. Hopefully, such formalizations will spawn the quantity and quality of theoretically oriented results that we have witnessed for concept learning.

Three basic formal models of concept learning have emerged as the most dominant. These learning protocols are certainly not the only ones that are relevant to the investigations appearing in the literature of computational learning theory. However, they do provide a starting point for understanding much of the recent theoretically-oriented research in concept learning.

The most traditional of these models, *identification in the limit*, has its origins in the work of Gold (1967). Much of the research within this model has focussed on determining the basic computational limitations on learning. A successful learning algorithm is characterized by its ability to exactly identify any target rule of a given type from examples *in the limit*, that is, as the number of examples grows arbitrarily large. Because computational feasibility has not been a central issue, many of the positive (learnability) results have relied on algorithms that are impractical with respect to time or space resources, and many of the negative (nonlearnability) results have been due to the fact that the learning domains have been too general to allow any algorithm to (ever) distinguish the target rule from among other possible hypotheses. Current research in this area, often employing powerful tools from the theory of recursive functions, continues to address a host of new and interesting philosophical issues in inductive inference; the reader is encouraged to consult the recent COLT proceedings for a sampling.

The two other central models of concept learning more closely address the question of what is meant by computationally efficient learning. One of these, upon which the first two papers in this issue are based, involves *exact identification in polynomial time* using various types of *queries*, and was recently introduced by Angluin (1988). For the purpose of understanding one variant of this model, we consider the problem of inferring an arbitrary propositional formula f over the Boolean variables $x_1, x_2, \ldots, x_n$. A learning algorithm may interact with its environment only in the following way: it may make an *equivalence query*, which is a question of the form "is g logically equivalent to f?," where f is the formula to be learned and g is the current hypothesis of the learning algorithm. If the answer is *yes* then the algorithm halts—it has succeeded at the inference task. Otherwise, the answer is *no* and the algorithm is given a counterexample to the correctness of g, that is, a truth assignment to the variables $x_1, \ldots, x_n$ such that the assignment satisfies f but not g, or vice-versa.

We demand that the algorithm is efficient, that is, that it runs in time polynomial in n (the number of variables, and hence the length of counterexamples) and in $|f|$ (the length of the representation of the formula f according to some reasonable encoding scheme). The class of Boolean formulas is said to be identifiable in polynomial time using equivalence queries if there is a single learning algorithm L and a polynomial p such that for any n and any formula f over n variables, L, using equivalence queries (which are answered with respect to f) runs in time at most $p(n, |f|)$, and outputs a formula g that is logically equivalent to f. The definition is easily extended to learning other types of concepts.

While the availability of an oracle to answer equivalence queries may seem too generous, in fact a number of arguments show that this learning protocol is a quite natural one for investigation (Angluin, 1988; Littlestone, 1987; Pitt, 1989). For example, if some class of concepts is identifiable in polynomial time using equivalence queries, then it is learnable in the distribution-independent model of Valiant (discussed below). Also, an algorithm that learns in polynomial time using equivalence queries can be used to obtain an algorithm that, in the absence of equivalence queries, classifies examples in an on-line setting and makes at most a polynomial number of errors of classification regardless of the stream of examples.

The first paper in this issue, by Angluin, tackles the basic question of whether various types of concepts can be learned in polynomial time using equivalence queries. As the title of her paper suggests, she shows for a variety of natural types of concepts that no polynomial time algorithm exists according to this protocol. Included are proofs that neither Boolean formulas in disjunctive nor conjunctive normal forms, regular languages, nor context-free languages are learnable in polynomial time using equivalence queries. Especially interesting is the fact that these nonlearnability results do not rely on any complexity-theoretic assumptions; in particular, they hold even if $P = NP$. The results are obtained via general combinatorial and information-theoretic techniques that may be useful in obtaining similar results for other types of concepts.

In light of these negative results, it is appropriate to ask what other information might be naturally available to a learning algorithm that would allow for successful inference. One way to increase a learning algorithm's power is to allow it to make *membership queries*—the algorithm may choose specific examples and ask whether they are classified as positive or negative examples by the target concept to be learned. This allows the algorithm to focus

its attention on examples that are near the *boundary* of the target concept. There are a number of recent results showing that various concept classes are learnable if membership queries are allowed in addition to equivalence queries. For these classes, nonlearnability using equivalence queries alone has been either established or is strongly suspected. Thus, for some classes of concepts, we are able to exactly determine the type of information that is necessary and sufficient for feasible inference.

The paper by Ishizaka is a result along these lines. He shows that there is a polynomial-time learning algorithm for the class of simple deterministic grammars using equivalence and membership queries.[1] The class of simple deterministic languages is a proper subclass of the context-free languages that properly includes the regular languages. An earlier result of Angluin (1987) shows that deterministic finite automata (that is, the regular languages) are learnable from equivalence and membership queries. Thus Ishizaka's result takes an important step in expanding the types of grammars (languages) for which efficient inference algorithms are known. Of particular interest is the fact that the algorithm necessarily involves a type of *constructive induction*: The language generated by the start symbol of a grammar is based on the sublanguages generated by the hidden nonterminal symbols of the grammar. Thus the learning algorithm must introduce new nonterminals that are not observed in interactions with its environment.

The third basic definition of concept learning that has received much attention is that of *probably approximately correct (PAC)* learning, introduced by Valiant (1984). As above, we use the class of Boolean formulas to explain the definition, which is easily generalized to other types of concepts. It is assumed that examples of the unknown target formula f are generated randomly according to an arbitrary and unknown probability distribution on example assignments of the n variables, and the algorithm is told whether each example satisfies f. The learning algorithm is also given parameters ϵ and δ, both less than 1. The algorithm is said to be successful at learning f if it outputs some formula g such that with probability at least $1 - \delta$, the likelihood that g incorrectly classifies the next randomly generated example is at most ϵ. The algorithm is required to run in time polynomial in n, $|f|$, $1/\epsilon$, and $1/\delta$. In short, this protocol allows us to specify a confidence parameter (δ), and a desired maximum classification error rate (ϵ), and a learning algorithm must ensure (with confidence $1 - \delta$) that the formula g learned has low distribution-weighted error (ϵ) relative to f.

The papers by Helmbold, Sloan, and Warmuth, and by Schapire, explore a variety of aspects of the PAC-learning model. Unlike the papers by Angluin and Ishizaka, which focus on the question of whether particular classes of concepts are learnable, these two papers are more concerned with general extensions and transformations of learning algorithms so as to improve either their scope of applicability or their performance. The paper by Helmbold, Sloan, and Warmuth gives a general technique for obtaining new learning algorithms for complicated concept classes based on learning algorithms for simpler ones. In particular, for concept classes C that are intersection-closed, they show how any PAC-learning algorithm can be easily extended to learn nested differences of concepts from C, that is, concepts of the form $c_1 - (c_2 - (\ldots - (c_{n-1} - c_n)))$, where each c_i is a concept from C that is learnable by the original algorithm. They show that their general transformation yields learning algorithms with near-optimal performance based on a variety of natural measures (for example, number of training examples needed). In doing so, the paper

demonstrates the utility of a number of techniques for the performance analysis of learning algorithms. This work may be especially interesting to the machine learning community, as a similar approach has independently been employed with considerable success in a practical setting (Salzberg, 1988).

Finally, in "The strength of weak learnability," Schapire proves the remarkable result that in order to achieve PAC-learning it is sufficient to find a learning algorithm that classifies examples only slightly better than random guessing. He gives a constructive technique for reducing the classification error rate of any learning algorithm whose success is independent of the input distribution of the examples. The construction may be iteratively applied to an algorithm which outputs hypotheses with a large error rate to obtain an algorithm outputting hypotheses with an error rate as small as desired, without increasing the time or sample complexity more than polynomially. The paper has a number of important implications regarding various aspects of learning. For example, the existence of a PAC-learning algorithm implies the existence of an on-line PAC-learning algorithm that is space-efficient in the sense that it saves relatively few of the examples that it has seen.

In summary, this issue of *Machine Learning* contains four papers that investigate, from a formal perspective, a variety of topics that are central to the theory of algorithmic concept learning. While I doubt that these few example papers are sufficient for the reader to learn the concept "interesting paper from COLT '89," I hope that they will provide an entry point from which to further pursue the many issues that arise in computational learning theory.

Leonard Pitt
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

Note

1. More accurately, his algorithm uses *extended equivalence queries*—consult the paper for details.

References

Angluin, D. (1987). Learning regular sets from queries and counterexamples. *Information and Computation, 75,* 87–106.
Angluin, D. (1988). Queries and concept learning. *Machine Learning, 2,* 319–342.
Gold, E.M. (1967). Language identification in the limit. *Information and Control, 10,* 447–474.
Haussler, D. and Pitt, L. (Eds.) (1988). *Proceedings of the 1988 Workshop on Computational Learning Theory.* San Mateo, CA: Morgan Kaufmann.
Littlestone, N. (1987). Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning, 2,* 285–318.
Pitt, L. (1989). Inductive inference, DFAs, and computational complexity. *Proceedings of AII-89 Workshop on Analogical and Inductive Inference; Lecture Notes in Artifical Intelligence 397,* (pp. 18–44), Heidelberg: Springer-Verlag.
Rivest, R.L., Haussler, D. and Warmuth, M.K. (Eds.) (1989). *Proceedings of the 1989 Workshop on Computational Learning Theory.* San Mateo, CA: Morgan Kaufmann.
Salzberg, S. (1988). *Exemplar-based learning: Theory and implementation.* (Technical Report TR-10-88). Cambridge, MA: Harvard University, Center for Research in Computing Technology.
Valiant, L.G. (1984). A theory of the learnable. *Communications of the ACM, 27,* 1134–1142.