

NASA-CR-194595

(NASA-CR-194595) REPRESENTATION IN
INCREMENTAL LEARNING
(Massachusetts Univ.) 4 p

N94-15719

Unclas

G3/80 0190814

N-80-CR
190814
NCC 2-658
4P

Representation in Incremental Learning

Department of Computer Science, Lederle Graduate Research Center
University of Massachusetts, Amherst, MA 01003

1 Introduction

Our work focused on two areas in machine learning: representation for inductive learning and how to apply concept learning techniques to learning state preferences, which can represent search control knowledge for problem solving. Specifically, in the first area we addressed the issues of the effect of representation on learning, on how learning formalisms are biased, and how concept learning can benefit from the use of a hybrid formalism. In the second area we examined the issues of developing an agent to learn search control knowledge from the relative values of states, of the source of that qualitative information, and of the ability to use both quantitative and qualitative information in order to develop an effective problem-solving policy.

2 Representation for Inductive Learning

The focus of our work in representation was on illustrating the benefits of hybrid concept representations (or formalisms) and on how to determine a good one automatically. Our research was motivated by the following two observations. Firstly, because an algorithm's concept representation space defines the space of possible generalizations, not even an exhaustive search strategy can overcome a poor choice of representation. In addition, for many data set, no one formalism will be best for the entire data set; one can do better by forming a hybrid representation.

Our work on hybrid formalisms commenced with an extension to the perceptron tree algorithm (Utgoff, 1989), which first tried a simple perceptron as the test; if it was doing poorly, it was replaced by a single variable test. Perceptron trees were designed to draw on the strengths of both decision trees and perceptrons thereby forming a more powerful representation language. However, perceptron trees permit perceptrons only at the leaves of the tree; all internal nodes are univariate symbolic tests. Our first extension to the perceptron tree algorithm was to create an incremental multivariate decision tree algorithm, PT2, that allows perceptrons at any node of the tree (Utgoff & Brodley, 1990). Each node in the tree is a linear threshold unit based on one or more features that describe each instance of the data. From this starting point, our research then split into two complementary directions of research: the first was an in-depth exploration of the issues in constructing multivariate decision trees and the second was a more general exploration of the strengths of hybrid formalisms and automatic algorithm selection.

We developed several multivariate decision tree methods to address the issues for constructing multivariate decision trees: representing a multivariate test, including symbolic and numeric features, learning the coefficients of a multivariate test, selecting the features to include in a test, and pruning of multivariate decision trees (Utgoff & Brodley, 1991a; Brodley & Utgoff, 1992). We performed an extensive empirical evaluation of our new methods and several well-known methods across a variety of learning tasks (Brodley & Utgoff, to appear). Our results demonstrated that some multivariate methods are generally more

effective than others under reasonable assumptions. In addition, the experiments confirm that allowing multivariate tests generally improves the accuracy of the resulting decision tree over a univariate tree.

The second direction of our work on representation was motivated by the observation that each inductive learning algorithm has a bias that may or may not be appropriate for a given learning task. The results of empirical comparisons of existing learning algorithms illustrate that each algorithm has a *selective superiority*; it is best for some but not all tasks. Given a data set, it is often not clear beforehand which algorithm will yield the best performance. Therefore we concluded that in such cases one must search the space of available algorithms to find that produces the best classifier. We developed an approach that applies knowledge about the representational biases of a set of learning algorithms to conduct this search automatically. In addition, the approach permits the model classes of the available algorithms to be mixed in a recursive tree-structured hybrid.

We implemented the approach in a system called the Model Class Selection System (MCS), which performs a heuristic best-first search for the best hybrid classifier for a set of data. Currently, MCS forms recursive hybrid classifiers using three primitive formalisms: decision trees, linear combination tests and instance-based classifiers. An empirical comparison of MCS to each of its primitive learning algorithms, and to the computationally intensive method of cross-validation, illustrated that automatic selection of learning algorithms using knowledge does indeed provide a solution to the selective superiority problem (Brodley & Utgoff, 1993; Brodley, 1993).

This research project is now funded by the National Science Foundation.

3 Learning State Preferences

In the second research area, we focused on the types of training information available to a learning agent in a problem-solving framework. In performing search, an agent need only determine the relative worth of competing states to decide which state to expand next. Equivalently, the agent need only know the state preferences in order to function effectively as a problem-solving agent. While developing methods to allow an agent to learn the state preferences, we observed that there are at least two types of training information available to the agent. Furthermore, these two types of information are not competitive, but complementary. The first form of information we call ‘quantitative’ because it helps to establish the exact values of states. The other, because it specifies the relative preference of certain states over others, is called ‘qualitative’.

Our work began with the development of evaluation functions for search control, employing both quantitative and qualitative information (Utgoff & Clouse, 1991b). The learning agent performed best-first search and employed quantitative information received from a form of temporal-difference (TD) learning in order to associate with each state the length of the shortest path to the goal.

The qualitative information was provided to the learning agent by an *expert* on the task, implemented as a complete search of the problem space. The expert would specify to the learning agent the proper state to expand in any given situation, thus providing a relative ordering of state preferences. At first we only compared the use of qualitative versus

quantitative information. Then, we observed that the two types of information could be melded into an integrated learning method.

In the integrated learning method, the learning agent used both forms of training information. The learner relied mostly on the quantitative information, but when certain conditions arose indicating that the agent was not benefiting from that information, the expert was invoked to provide an instance of qualitative information. In comparing the two sources separately and then together, we found that being able to take advantage of both forms of information allowed a learning agent to learn to perform the task more quickly in combination than with either single source alone.

Although the combination of the two sources of information allowed the learner to develop an appropriate search strategy more quickly than if it had only seen one of the forms of information, providing qualitative information through a complete search of the space is not feasible for most problems. Therefore, we focused on finding a better source for qualitative information.

This led to a method that allows a *human* to provide qualitative information to a learning agent (Clouse & Utgoff, 1992). As the agent is learning to perform a task and is relying on a reinforcement learning method to provide it with quantitative information, a human observes the performance of the learning agent on the task. The goal of the human is to teach the learning agent by providing the agent with qualitative information in the form of an action that the agent should take at a particular time.

At each time step, the learning agent learns either from the qualitative information provided by the human or from the quantitative information provided by the reinforcement method. Deciding which form of information to use is based solely on the presence of that information: if the human has specified an action to take, the learning agent uses that action and learns that that action is the appropriate one in the situation in which it was given. If the human has not provided any qualitative information, the agent relies only on the reinforcement method for training information.

In experiments with the classic cart-pole task, the agent that employed both forms of information was able to learn to balance the pole in two orders of magnitude fewer trials than an agent using only the quantitative information. In another more complicated task, the number of trials needed to achieve success was reduced by more than one order of magnitude.

Thus, we have found that allowing a learning agent to employ both quantitative and qualitative information about its task is advantageous. In a proposal submitted to the AFOSR, we have proposed to address several of the central issues raised by this conclusion. For example, we want to determine analytically why the integration of the two forms of information produces better learners, and we want to develop new methods for obtaining qualitative information from human experts. Also, given an understanding of the integration, we will become able to build learning systems that take better advantage of the synergy of the two forms of information.

References

- Brodley, C. E., & Utgoff, P. E. (1992). *Multivariate versus univariate decision trees*, (Coins Technical Report 92-8), Amherst, MA: University of Massachusetts, Department of Com-

puter and Information Science.

- Brodley, C. E., & Utgoff, P. E. (1993). Dynamic recursive model class selection for classifier construction. *Preliminary Papers of the Fourth International Workshop on Artificial Intelligence and Statistics* (pp. 179-184).
- Brodley, C. E., & Utgoff, P. E. (to appear). Multivariate decision trees. *Machine Learning*.
- Brodley, C. E. (1993). Addressing the selective superiority problem: Automatic algorithm/model class selection. *Machine Learning: Proceedings of the Tenth International Conference* (pp. 17-24). Morgan Kaufmann.
- Clouse, J. A., & Utgoff, P. E. (1992). A teaching method for reinforcement learning. *Machine Learning: Proceedings of the Ninth International Conference* (pp. 92-101). San Mateo, CA: Morgan Kaufmann.
- Utgoff, P. E. (1989). Perceptron trees: A case study in hybrid concept representations. *Connection Science*, 1, 377-391.
- Utgoff, P. E., & Brodley, C. E. (1990). An incremental method for finding multivariate splits for decision trees. *Proceedings of the Seventh International Conference on Machine Learning* (pp. 58-65). Austin, TX: Morgan Kaufmann.
- Utgoff, P. E., & Brodley, C. E. (1991a). *Linear machine decision trees*, (COINS Technical Report 91-10), Amherst, MA: University of Massachusetts, Department of Computer and Information Science.
- Utgoff, P. E., & Clouse, J. A. (1991b). Two kinds of training information for evaluation function learning. *Proceedings of the Ninth National Conference on Artificial Intelligence* (pp. 596-600). Anaheim, CA: MIT Press.