UNCERTAINTY BASED SELECTION OF LEARNING EXPERIENCES

Paul D. Scott[1] **& Shaul Markovitch**
Center for Machine Intelligence
2001,Commonwealth Blvd.,Ann Arbor, Michigan 48105

ABSTRACT

The training experiences needed by a learning system may be selected by either an external agent or the system itself. We show that knowledge of the current state of the learner's representation, which is not available to an external agent, is necessary for selection of informative experiences. Hence it is advantageous if a learning system can select its own experiences. We show that the uncertainty of the current representation can be used as a heuristic to guide selection of experiences, and describe results obtained with DIDO, an inductive learning system we have developed using an uncertainty based selection heuristic.

STRATEGIES FOR SELECTING EXPERIENCES

It has long been recognized that the speed with which a system learns is strongly influenced by the particular selection of experiences, from the space of all possible experiences, that the system receives. For example, Winston (1975) showed that 'near misses' (training examples that lie just outside the boundaries of the concept to be learned) greatly facilitate the speed with which his program learns new concepts. Some experiences will lead to major changes in a learning system's representation of its domain, others will lead only to minor changes, while the rest will produce no change at all. Experiences that lead to representation changes are called *informative experiences*. The goal of experience selection is to produce as high a proportion of informative experiences as possible.

On many machine learning systems, experience selection is carried out by an external agent, usually called a teacher[2]. However such an external agent does not have all the knowledge needed to select the most informative experiences. How informative a particular experience will be to a learning system depends upon both the domain about which the system is learning, and the current state of the system's representation of that domain. An external agent may have knowledge of the domain, but does not normally have direct access to the current state of the representation. The importance of the latter is demonstrated by the results shown in Table 1 which were obtained by selecting training examples for the candidate elimination algorithm (Mitchell, 1977) using three simple strategies. The *Random* (examples selected completely at random) and *Balanced* (equal numbers of positive and negative examples selected at random) strategies do not use knowledge of the current representation. In contrast, the *Asking* strategy selects random examples of concepts which have not yet been eliminated from the version space. Examples were classified as informative if they resulted in a reduction of the version space. The means given were the average of 50 runs with different target concepts (See Scott 1989 for further details). These results demonstrate that making use of knowledge of the current representation can drastically reduce the number of experiences needed to learn a concept by increasing the proportion of informative examples selected.

TABLE 1

---

[1]Some of the work reported in this paper was supported by NSF Grant # MCS-8203956.

[2] Teachers may provide other forms of assistance in addition to experience selection. In particular very many inductive concept learning systems depend upon a teacher to classify experiences.

	Random	Balanced	Asking
Mean number training examples	207.3	123.9	14.7
Mean number informative examples	19.9	8.4	12.4
Mean % informative examples	9.6	6.7	84.3

In order to select a high proportion of informative experiences, an external agent must acquire knowledge of the learning system's current representation. The only method by which this can normally be done is to derive a model of the learning system's representation from its observable behavior. This is a learning task which is likely to be more difficult than the original learning task. Thus depending on an external agent to select experiences would seem to require an external agent that is a more effective learning system than the learning system that it is helping. This condition is satisfied when a human teacher supplies experiences to a comparatively simple machine learning program. However, the availability of such a 'smart' teacher is not something that can always be assumed.

Instead of depending on an external agent a learning system could select its own experiences. Unfortunately, the system itself does not have all the knowledge necessary to make an optimal selection. It may have knowledge of the current state of its representation but it lacks knowledge of the domain (otherwise there would be nothing for it to learn). For example, it is not possible to build a learning system which selects only 'near misses' since that would require knowledge of the target concept it is attempting to learn. As the results reported above show, in some circumstances, the system's ignorance of the domain need not be a problem. Unfortunately the *Asking* strategy as described is restricted to operating in conjunction with the candidate elimination algorithm, and hence subject to the same restrictions on domain of applicability: learning conjunctive concepts from noise free examples in a small space.

It therefore seems worthwhile to consider whether it is possible to devise a method for selecting informative experiences that is independent of the method used to derive a representation from those experiences. It is important to note that the system's ignorance of the domain is not total. Initially it may know very little but as learning progresses it acquires a representation which contains information about the domain. Is it possible to devise a simple domain independent method of assessing how informative a potential experience is likely to be by using the representation which has been acquired of previous experiences?

UNCERTAINTY AS AN EXPERIENCE SELECTION HEURISTIC

In order to select its own experiences, a learning system requires some form of evaluation function which enables it to assess the likelihood that members of a particular class of potential experiences will be informative. Such an evaluation function would serve as an *experience selection heuristic*. Informative experiences must satisfy two criteria: they must provide new information about the domain, and it must be possible for the learning system to relate them to what is already known about the domain.

The first criterion could be satisfied using *novelty* as an experience selection heuristic: given a choice, the experience least like any previous experience is selected. This selection strategy would generate a stream of highly dissimilar experiences and hence lead to the development of a very shallow and highly fragmented representation of the domain. Most new experiences would be uninformative because they could not be related to previous experience. The converse heuristic is *familiarity*: the experience most like previous experiences is selected. This would satisfy the second criterion. Unfortunately it would generate a stream of very similar experiences. Hence most new experiences would be uninformative because they would simply confirm what the system had already learned. Furthermore, the system learning would be confined to a small area of the domain. The novelty heuristic will produce a pathological generalist while the familiarity heuristic will produce an equally pathological specialist.

What is needed is a compromise between these two extremes. Such a compromise would be provided by a heuristic that selected experiences that provided evidence about aspects of the current representation which were uncertain. A heuristic with this characteristic would generate experiences which lay close to the boundary between what the learning systems does and does not know. There is a well established measure of uncertainty: the Shannon uncertainty function (Shannon & Weaver, 1949) which is defined as

$$H = -\sum_{i=1}^{n}(p_i \times \log_2(p_i))$$

where the sum is taken over a set of probabilities. This uncertainty measure can be used with any representation whose changeable elements have a finite (though not necessarily fixed) set of alternative values, and in which

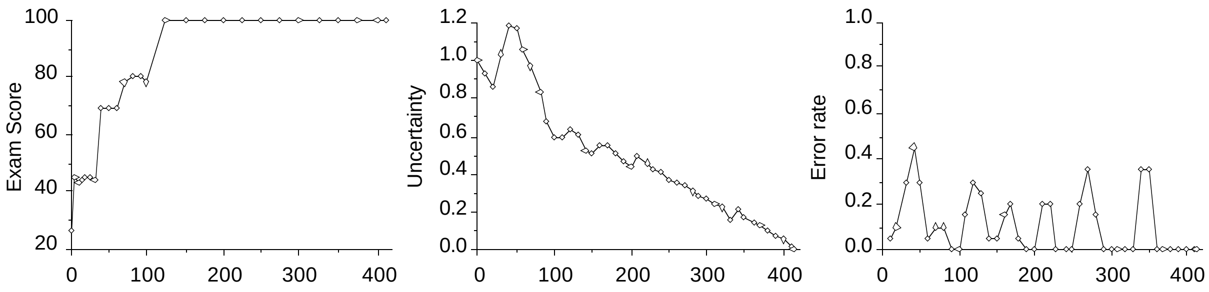

Figure 1. DIDO's performance during a typical run plotted as functions of number of training experiences.

probability estimates can be derived the correctness of those alternatives. In the next section we describe DIDO, an inductive learning program which uses the uncertainty heuristic to select its own experiences.

UNCERTAINTY BASED EXPERIENCE SELECTION IN DIDO

DIDO (Scott & Markovitch 1988, 1989) is an inductive learning program we have developed which uses the uncertainty heuristic to select its own experiences. The goal of the program is to reach a state in which it can predict the outcomes of applying operations to all entities encountered in its domain. DIDO builds a class inheritance hierarchy to represent what has been discovered about the application of actions to entities. Associated with each action in a class is a set of outcomes which the system has observed, together with a probability estimate for each outcome based on the frequency with which it has been experienced. Hence by applying the Shannon uncertainty function to this set of probabilities, the program can determine how uncertain it is of the outcome of applying the action to a member of the class. This value is computed for every action in every class. DIDO then generates a new experience by applying the action with highest uncertainty to a randomly selected member of the class in which that uncertainty occurs. The procedure by which DIDO uses evidence from experiences to modify and extend the representation is basically a form of learning by specialization. A description appears in Scott & Markovitch (1989). The effect of the uncertainty based experience selection heuristic is that DIDO steadily attempts to eliminate uncertainty from the representation of the domain.

Figure 1 shows some of the results obtained during a typical run of DIDO in which the program learned to predict the consequences of applying 3 different actions to a variety of types of object. A network of 9 classes was needed to provide a complete representation of this domain which involved both conjunctive and disjunctive concepts.

How successful is DIDO at learning to predict the outcome of applying actions to objects? The left hand graph in Figure 1 shows how DIDO's ability to make correct predictions rose during learning. This data was obtained by periodically stopping the learning process and presenting DIDO with an 'exam'. Exams comprise 100 randomly selected action-object pairs. DIDO is required to predict the outcome of applying the action to the object. As can be seen, the learning procedure is effective and DIDO correctly predicts the outcome of all the examples presented in the exams after 120 training experiences.

Does DIDO's uncertainty about the domain decline steadily during learning? The middle graph shows the average uncertainty, taken over all actions in all classes, during learning. At a given time most uncertainties in the system have approximately the same value since DIDO's learning strategy is based on reducing the currently highest uncertainty in the representation. This graph demonstrates steady progress in eliminating uncertainty.

Does the uncertainty heuristic produce informative experiences? If it does then one would expect the experiences selected to be those about whose outcome DIDO is still doubtful. The right hand graph shows the error rate which provides a measure of this. The error rate is the proportion of *selected experiences* that DIDO predicts incorrectly and is shown as averages taken over successive 20 experience intervals. As can be seen, it is erratic but shows no tendency to decline permanently until the final stage of learning. We have already seen from the exam score results that DIDO is able to correctly predict the overwhelming majority of *possible experiences* for most of the learning period. Hence the error rate results show that the uncertainty heuristic is selecting experiences from the very small subset about which DIDO has not yet learned. Thus the uncertainty heuristic does indeed select informative experiences throughout learning.

DISCUSSION

We have run experiments with DIDO in a variety of domains, including some which were non-deterministic. These have shown that, in conjunction with the particular representation and learning method used by DIDO, uncertainty provides a robust and effective experience selection heuristic. Would the heuristic be equally effective in conjunction with other representations and learning methods? Its use is confined to those representations which retain information about the frequency with which types of events have occurred, but it seems reasonable to extend representations which lack this feature to provide support for the experience selector. Further studies to investigate how well the heuristic operates with other inductive learning methods would be of great interest. Other learning methods may require non-statistical measures of uncertainty. The experience selector in PRODIGY (Carbonell & Gill, 1987), which operates in conjunction with an explanation based learning method, generates relevant new experiences when it encounters a contradiction between its representation and an experience. This is equivalent to selecting new experiences pertinent to any non-zero uncertainty.

Other learning programs which generate their own experiences have successfully used heuristics not based on some measure of uncertainty. LEX (Mitchell, Utgoff & Banerji, 1983) uses a method that is essentially refinement of the Asking strategy described earlier and appears to be tied to a version space type of representation. Lenat's AM (Lenat 1982) uses a highly complex experience selection heuristic, which he terms 'interestingness', composed of 54 heuristic evaluation rules. It is unclear what the relative contribution of these rules was and how domain specific they were (Ritchie & Hanna, 1984).

Experience selection is clearly an important aspect of the learning process upon which comparatively little work has been done. It is however only one of several selection processes which can occur at various stages throughout the learning process. Our recent work has led us to the conviction that the study of such selection processes will play a central role in the further development of machine learning (Markovitch & Scott, 1989).

References

Carbonell,J. & Gill,Y., 1987, *Learning By Experimentation*, Proc. 4th International Workshop on Machine Learning, UC Irvine, June 1987.

Lenat,D. 1982, *AM: Discovery in Mathematics and Heuristic Search* . In Davis,R. & Lenat,D.B. , *Knowledge-based Systems in Artificial Intelligence*, McGraw-Hill, New York.

Markovitch,S. & Scott,P.D. , 1989, "*Information Filters and Their Implementation in the Syllog System*", Proc. 6th International Workshop on Machine Learning. (Also available as CMI Tech Report CMI-89-006).

Mitchell,T.M., 1977, *Version Spaces: A Candidate Elimination Approach to Rule Learning*, Proceedings Fifth International Joint Conference on Artificial Intelligence, Cambridge, Mass. pp 305-310.

Mitchell,T.M., Utgoff,P.E. & Banerji,R., 1983, *Learning by Experimentation: Acquiring and Refining Problem Solving Heuristics.* In *Machine Learning* eds Michalski, Carbonell & Mitchell, Tioga Press (1983).

Ritchie,G.D. & Hanna,F.K., 1984, *AM: A Case Study in AI Methodology*, Artificial Intelligence **23** pp 249-268.

Scott,P.D., 1989, *When Is Asking Better Than Being Told?*, Center for Machine Intelligence Technical Report CMI-89-012.

Scott,P.D. & Markovitch,S., 1989, *Learning Novel Domains Through Curiosity and Conjecture*, Proc. IJCAI-89, 11th International Joint Conference on Artificial Intelligence. (Also available as CMI Tech Report CMI-89-010).

Scott,P.D. & Markovitch,S., 1988, *Learning Through Curiosity and Conjecture*, CMI Tech Report #030988.

Shannon ,C.E& W.Weaver, 1949, *The Mathematical Theory of Communication*, University of Illinois Press.

Winston,P.H., 1975, *Learning Structural Descriptions from Examples* , In *The Psychology of Computer Vision* ed P.H.Winston, McGraw-Hill.