LEARNING IN MASSIVELY PARALLEL NETS

Geoffrey E. Hinton

Computer Science Department
Carnegie-Mellon University

Extended Abstract

The human brain is very different from a conventional digital computer. It relies on massive parallelism rather than raw speed and it stores long-term knowledge by modifying the way its processing elements interact rather than by setting bits in a passive, general purpose memory. It is robust against minor physical damage and it learns from experience instead of being explicitly programmed. We do not yet know how the brain uses the activities of neurons to represent complex, articulated structures, or how the perceptual system turns the raw input into useful internal representations so rapidly. Nor do we know how the brain learns new representational schemes. But over the past few years there have been a lot of new and interesting theories about these issues. Much of the theorizing has been motivated by the belief that the brain is using computational principles which could also be applied to massively parallel artificial systems, if only we knew what the principles were.

In the talk, I shall focus on the issue of learning. Early research on perceptrons and associative nets (or matrix memories) showed how to set the weights of the connections between input units and output units so that a pattern of activity on the input units would cause the desired pattern of activity on the output units. A variant, called the auto-associative net, did not distinguish between input and output units. It modified the weights of pairwise inter-connections among the units to ensure that any sufficiently large part of a stored pattern could recreate the rest. Recently, Hopfield has developed an interesting way of analyzing the behavior of iterative, auto-associative nets, but research on simple associative networks is generally of limited interest because most interesting tasks are too complex to be performed by auto-association or by direct connections from the input units to the output units. Many intervening layers of "hidden" units are generally required and the tough learning problem is to decide how to use these hidden units. The reason this is so difficult is that we are requiring the network to invent its own representational scheme, and the space of possible schemes is immense, even if we restrict ourselves to those that can be implemented conveniently in networks of neuron-like units.

For many years there was little progress in developing learning schemes that were powerful enough to construct sensible representations in the hidden units. But in the last few years, many different methods have been invented. Some of these use gradient-descent in weight space: They slowly adjust the weights of the connections among the hidden units in such a way that the errors produced by the whole network are progressively reduced. Gradient descent procedures like the Boltzmann machine learning procedure or the back-propagation learning procedure can construct surprisingly subtle representations. Examples are given in Rumelhart and McClelland, 1986 or Saund (this proceedings). They often create distributed representations in which important entities are represented by the pattern of activity in a set of units rather than by activity in a single unit. Unfortunately, these gradient descent procedures do not scale well. With more than a few thousand connections they learn extremely slowly. They are also not very plausible as models of learning in the brain.

The current challenge is to find ways of making gradient descent procedures scale properly or to find ways of organizing multilayer unsupervised or semi-supervised procedures in which the units are "trying" to achieve some more local goal than minimizing the overall error of the network. For example, it is possible to modify a unit's weights in such a way that the output of the unit becomes more informative about which of several distributions the input came from. This can be done without prescribing just how the unit should behave. It could choose to become active only for a few cases drawn from one distribution and never otherwise, or it could choose to come for many cases drawn from one distribution and for fewer cases drawn from other distributions. Preliminary experiments suggest that this kind of partial supervision has much better scaling properties than stricter supervision.

One promising class of learning procedures is based on a reinforcement paradigm Barto, 1985. These procedures, which are closely related to theories of evolution or learning automata, can be very simple and yet they can learn interesting representations. One new procedure in this class will be described in some detail during the talk. Because it is based on reinforcement, it may allow good scaling to be achieved by allowing modules to set up reinforcement schedules for other modules. If this can be done it will bring this whole area of research much closer to conventional artificial intelligence in which complexity is typically handled by using hierarchies in which modules at one level set up subgoals (i.e. reinforcement schedules) for modules at the next level down.

Barto, A. G. Learning by statistical cooperation of self-interested neuron-like computing elements. *Human Neurobiology*, 1985, **4** 229-256.

Rumelhart, D. E. McClelland, J. L. & the PDP research group *Parallel distributed processing: Explorations in the microstructure of cognition.* MIT Press: Cambridge Mass. 1986.