Recurrent Expert Networks

Cathie LeBlanc
Dept of Computer Science
Florida State Univ
Tallahassee, FL 32306-4019
leblanc@cs.fsu.edu

Research has shown that computational techniques such as neural networks often provide classification abilities that are more accurate than methods which rely on explicit knowledge acquisition alone (Ben-David & Mandel 1995). On the other hand, because no "reason" for a particular classification can be given when a computational technique has been used, human experts tend to be skeptical of such systems. As a result, many researchers have developed tools, called hybrid systems, which combine the pattern recognition capabilities and parallel processing of neural systems while retaining the domain knowledge encoded in expert systems (Medsker 1994).

Because the widely known "knowledge acquisition bottleneck" makes explicit knowledge acquisition tools (such as expert systems) expensive to create, Kuncicky, Hruska and Lacher (Kuncicky, Hruska, & Lacher 1992) have developed expert networks, which eliminate the need for the expert to associate a certainty factor with each rule. Instead, the expert system rules, acquired from the human expert, are translated into the topology of a computational network, called an expert network. The individual nodes in an expert network are not all identical but instead have functionalities that match the part of the knowledge base which they encode. For example, a node which encodes an AND between two pieces of knowledge in the knowledge base might take the minimum of its inputs and if that minimum is above a certain threshold, outputs that minimum (Lacher & Nguyen 1994). The certainty factors correspond to the trainable weights between the nodes. Example data are presented to the network and the weights are learned via a backpropagation-like algorithm (Lacher, Hruska, & Kuncicky 1992). The topologies and learning algorithms developed for expert networks thus far have been strictly feed-forward.

Some classification tasks, however, will be difficult to complete using a strictly feed-forward architecture. In particular, the solution to many problems requires that "state" information be maintained. State information is the context in which the problem is currently being solved. The context of a problem solution will change as the solution proceeds. Certain sets of rules need only be considered in certain contexts. For example, if the problem is to read email, the state must include information about whether the computer is on or off. The rules to turn on the computer will only be considered if the state tells us that the computer is off.

Such state information will be difficult to manage using feed-forward architectures. In fact, in standard artificial neural networks, such state information is handled by the addition of recurrent connections in the topology of the network (Hertz, Krogh, & Palmer 1991). The recurrent connections allow the context information to be input to the network at succeeding steps. Therefore, I will extend the notion of expert networks so that they will be able to maintain state information via recurrent connections while at the same time encoding previously discovered expert knowledge.

The problem domain to which I will apply this technology is the protein folding problem. In this problem, state information about secondary structure predictions for amino acids earlier in the protein's primary sequence will play an important role in the secondary structure prediction for the current amino acid.

References

Ben-David, A., and Mandel, J. 1995. Classification accuracy: Machine learning vs. explicit knowledge acquisition. *Machine Learning* 18:109–114.

Hertz, J.; Krogh, A.; and Palmer, R. G. 1991. *Introduction to the theory of neural computation.* Redwood City, California: Addison-Wesley Publishing Company.

Kuncicky, D.; Hruska, S.; and Lacher, R. 1992. Hybrid systems: the equivalence of expert system and neural network inference. *International Journal of Expert Systems* 4:281–297.

Lacher, R., and Nguyen, K. 1994. Hierarchical architectures for reasoning. In Sun, R., and Bookman, L., eds., *Computational Architectures for Integrating Neural and Symbolic Processes*, 117–150. Boston: Kluwer Academic Publishers.

Lacher, R.; Hruska, S.; and Kuncicky, D. 1992. Backpropagation learning in expert networks. *IEEE Transactions on Neural Networks* 3:62–72.

Medsker, L. R. 1994. *Hybrid Neural Network and Expert Systems.* Boston: Kluwer Academic Publishers.