Neural Methodologies in Rule-Based Expert Systems

E. Burattini and G. Tamburrini
Istituto di Cibernetica C.N.R.
I-80072 Arco Felice
e-mail: ernb@sole.cib.na.cnr.it

1 Introduction

Early research in artificial neural networks (ANN's) was significantly influenced by the goal of providing simplified models of neural cells in the brain and their mutual interactions. The related goal of investigating whether perceptual or more properly cognitive behaviours could be simulated by suitably organized ANN's was initially motivated by the heuristic idea that similarity of *structure* between artificial and real neural nets could provide clues for isolating similarities of *function*. The programmatic goals of Artificial Intelligence, whose inception as an independent discipline dates back to the mid 1950's, mark a rather sharp departure from this heuristic strategy: computational implementation of human-level cognitive functions can be pursued independently of the study of brain structure. Expert system research is a special area of AI which inherits this methodological assumption, and is further constrained by the goal of arriving at computational systems matching the performance of human specialists in their domain of expertise. Since no additional hypotheses are made on the methods to be used to achieve this goal, neural methodologies are, from this perspective, on a par with any other computational method: their usefulness in expert systems is to be assessed on the basis of appropriate performance criteria. In this respect, the mere advertisement of learning and parallel processing capacities of ANN's is too generic, and sometimes even misleading: consideration of the known (theoretical or contingent) limitations of these capacities –*vis--vis* the real demands for expert system design and implementation– is also needed to decide when purely neural, purely symbolic, or hybrid methodologies are to be adopted. Initial efforts in this direction are Gutchnecht and Pfeifer (1990), Steels (1989). The present paper is meant as a contribution towards this sort of assessment in the more specific area of *rule-based* expert systems. It presents, in a more systematic fashion, the broad methodological options underlying technical work in this area by the present authors and other members of the same research group (see, for example, Burattini *et al.* (1992, 1995), Aiello *et al.* (1995a, 1995b), De Gregorio (1994, 1996)), in addition to a selective comparison with other approaches.

2 Neural learning for knowledge and data acquisition

A crucial problem encountered when designing an expert system is that of identifying and appropriately codifying domain knowledge and problem solving strategies of human specialists. How is this task to be achieved? Puppe (1993) distinguishes between three different modes of knowledge acquisition: *indirect, direct,* and *automatic.* Indirect knowledge acquisition requires the work of a special character, a "knowledge engineer" extracting knowledge by communication with human experts. This is the earliest and, notwithstanding its shortcomings, the still prevailing approach to knowledge acquisition. It is time-consuming, since long interactions of two groups of specialists are needed. It is not immune from errors, especially because the two groups do not share the same background. If human experts possess the ability to render fully explicit and formalize their knowledge, then direct knowledge acquisition becomes possible, thus reducing the impact of errors from failure of inter-group communication: the experts are responsible for the construction, testing, and updating of the expert system. But the confluence of these abilities in the same individual is rather uncommon. The third mode, automatic knowledge acquisition by means of learning methods, *whether involving traditional symbolic programming or neural network training,* is presently only a remote possibility:

> Unfortunately, automatic knowledge acquisition turned out to be extremely difficult, which is not surprising in the face of the following considerations: learning requires knowledge; the more one knows, the easier is to learn, and human experts need about ten years of intensive occupation with their field before they become experts, whereby, unlike programs, they can build on general knowledge. Even learning programs therefore require complex knowledge acquisition. Perhaps even more serious is the fact that the already difficult validation problem is becoming even more acute. When a human being constructs or alters a knowledge base himself, he can estimate the performance of the expert system much better than when a learning program makes changes of its own accord. To put it another way, who would take the responsibility for automatically generated knowledge for expert systems used in practice? [see Puppe (1993), 9-10]

Even though the prospects of fully automatizing knowledge acquisition seem to recede in a rather distant future, learning algorithms may still be used as a supporting tool for knowledge engineers. In order to highlight current limitations of neural network learning in this area, let us consider the scaling up for a toy problem discussed in Gallant (1993), p. 267f. The problem is that of constructing a neural expert system capable of deciding which one of two possible diseases affects patients presenting pathological patterns from six possible symptoms. Furthermore, the system has to prescribe appropriate therapies. A neural network is trained to execute this task. The network, a multilayer perceptron, is formed by three layers of neurons. Each neuron in the first (input) layer represents a symptom. The second layer is formed by two neurons representing the diseases the system has to discriminate between. Each neuron in the third layer represents a possible therapy. This network is trained by means of a set of pathological patterns. Each pattern is associated to a disease and a therapy, and the connections between the elements of the network are modified so as obtain the right associations between initial pathological patterns on the one hand, and disease and therapy on the other hand. If the learning procedure does not converge to a correct classification, then more neurons are added to the network until the correct classification is obtained. Is the training on this miniature knowledge base generalizable to interesting situations? A prospective physician is trained over hundreds of medical cases, in the light of previously acquired theoretical knowledge. Approximating this learning process, even in the limited sense of collecting, organizing, and presenting a network with a comparable number of significantly different cases, is a tremendously time-consuming task. Moreover, medical domains in which a few hundreds of *diseases* and *pathological patterns* are to be taken into account are rather common. The computational complexity of learning for problems of these dimensions is a serious issue (Judd (1990)), and casts doubts on this approach to constructing large knowledge bases.

In view of these considerations, it is reasonable to suggest that expert system research can take advantage of learning algorithms for constructing knowledge bases mainly where human expertise consists of a classification of patterns into a *small number* of different classes. But even within these boundaries, some qualifications are appropriate. Classification by human experts may involve elaborate, *stepwise reasoning,* and providing a justification for the conclusions that have been reached in this way, under the form of a report of inferential steps from input data, is a crucial demand for expert systems. However, the NN's trained by learning procedures to provide certain i/o associations do not usually lend themselves to this sort of intermediate state analysis: the hidden layers of neurons are semantically "opaque". And what about the even more restricted type of problems concerning one-step classifications of patterns into a *small number* of classes? The work of M. De Gregorio described in this issue of *Mathware* illustrates some of the difficulties which may be encountered there. He addressed the problem of classifying the shape of house portals starting from images which are part of the total input to an expert system for the overall classification of architectural styles in ancient Italian buildings. The markedly different spatial spreading of the relevant classes of portal shapes proved a formidable obstacle for the training of a multidiscriminator weightless NN, and suggested the opportunity of using a *hybrid* classifier, where the weightless NN recognizes geometrical features from portal contours. This information plays the twofold role of providing clues to an hypothesis formation reasoning module (specified as a production system), as well as data for corroborating the correctness of hypotheses on overall shape identification. Further, De Gregorio (1994) examined another visual classification problem (levels of traffic congestion from actual photographs of road junctions) into a small number of classes, and showed that multidiscriminator weightless NN's do not work substantively better than purely numerical algorithms. To strike a more optimistic note at the end of this section, let us mention a successful example of one-step classification into a small number of classes: the ECG signal classifier implemented by means of a weightless NN, which makes use of a clever segmentation of ECG traces (see Badr (1993)).

3 Parallelism in qualitative reasoning

Let us now turn to consider the heart of rule-based expert system, which is typically formed by a *knowledge base* codified as a system of production rules and a *rule interpreter* which works iteratively in recognize-and-act cycles. The latter may be used to implement various kinds of searches (e.g. forward or backward chaining, or mixed strategies). As is well-known, *propositional* production rules naturally lend themselves to parallel processing. In previous papers (see Burattini *et al.* (1992), Aiello *et al.* (1995a, b)), we have described a localist neural model which makes the parallel processing of such production systems possible: this neural model performs forward chaining, is capable of querying external sources about missing information needed to establish its goals, and can provide a detailed justification for the conclusions it has reached, by exhibiting a trace of the inferential steps leading from known data to conclusions. The localist semantics adopted (each literal is represented by a distinguished neuron) and the temporal dimension of activation spreading are exploited in making this trace available.

The production *rules* we have considered take the form

$$p_1 \wedge \ldots \wedge p_k \rightarrow c$$

where $p_1, \ldots, p_k$ and c are propositional literals. We used nets of the weighted-sum, thresholded neurons, introduced by Caianiello, for representing such rules. The state equation for a neuron $\mathbf{h}$ of this type is given by

$$u_h(t+1) = \mathbf{1}\left[\sum_{j=1}^{n}\sum_{i=0}^{t} a_{j,h} \cdot u_j(i) \cdot \delta_h(t-i) - s_h\right] \quad (1)$$

where $u_h(i)$ is the state (1 or 0) of the neuron $\mathbf{h}$ at time i; $a_{j,h}$ is the weight, or coupling coefficient between neurons j and h; $\delta_h(i)$ is a monotone non-increasing function of the discrete time i for neuron $\mathbf{h}$ representing a time-variable memory of the excitation received by $\mathbf{h}$; $s_{\mathbf{h}}$ is the threshold of $\mathbf{h}$; finally,

$$\mathbf{1}[x] \quad \text{is} \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases}$$

A rule r

$$p_1 \wedge \ldots \wedge p_k \rightarrow h$$

can be represented as a net having k neurons $p_1, \ldots, p_k$ connected to a neuron $\mathbf{h}$ (see fig.1) with the following settings:

$$\begin{aligned} a_{j,k} &= 1 \quad (1 \leq j \leq k) \\ s_h &= k - \epsilon \quad (0 < \epsilon < 1) \\ \delta_h(i) &= \delta^0(i) \text{ where } \delta^0 \text{ is } \begin{cases} 1 & \text{if } i = 0 \\ 0 & \text{if } i \neq 0 \end{cases} \text{ (i.e. there is no memory).} \end{aligned} \quad (2)$$

Fig. 1: Neural rule model

By (1) and the settings in (2), one has that

$$u_h(t+1) = 1 \quad \text{iff } \forall j \, u_{p_j}(t) = 1$$

Using this representation of rules as basic building block, one can design a purely neural model of forward chaining on systems of propositional production rules. Such systems, unlike, e.g., the systems treated in Towell and Shavlik (1994), may contain cycles: a literal appearing in the right-hand side of a rule can appear in the left-hand side of another rule. Moreover, several rules may share the same right-hand side literal. This latter possibility requires the introduction of a slight complication, with respect to the scheme in fig. 1, for the neural implementation of forward chaining: if a literal p occurs in the right-hand side of m distinct production rules, then m distinct neurons –each one representing an occurrence of p in these rules– is to be introduced. This condition is needed to avoid an incorrect activation of a neuron representing p, which may obtain from a combination of premises belonging to different rules. (See Burattini *et al.* (1995, pp. 11-12) for a specific example of such situation.)

Various interpretations appropriate for expert system inferencing can be given of the formalism of production rules. In addition to the immediate logical interpretation in terms of propositional Horn clauses (where only positive literals occur), the arrow can be taken to stand for causal relationships involved in forms of uncertain qualitative inference, such as the *abductive* and *predictive* causal inferences used in diagnostic reasoning for hypothesis formation and testing (see, e.g., Burattini and De Gregorio (1994)). Roughly, abductive causal inferences enable one to select possible explanatory hypotheses for (causes of) observed facts, and predictive causal inferences enable one to isolate possible observable manifestations of the explanatory hypotheses selected by abductive inferences. For example, interpreting '$\Phi_1 \to \Psi$' for 'Φ_1 is possibly caused by Ψ', and knowing that Φ_1 is the case, one can hypothesize (*abduce*) Ψ as an explanation for occurrence of Φ_1. Similarly, if '$\Psi \to \Phi_2$' is taken to mean 'Ψ may give rise to manifestation Φ_2', then, from Ψ one may predict that Φ_2 is likely to be observed.

On the basis of this model of forward chaining on propositional Horn clauses and (two-level) causal nets, purely neural, rule-based diagnostic systems for paediatric gastroenterology and the static analysis of buildings have been designed (see Aiello *et al.* (1995b) and references therein). The attribute 'purely neural' indicates that in addition to knowledge base and rule interpreter, the control of inferential processes is modeled by suitably organized NN's. And domain-specific heuristics (for, e.g., choosing between competing diagnoses) have been neurally modeled. The parallel processing of rules made possible by these models may make a difference in diagnostic domains where reaction times are crucial parameters for evaluating the performance of expert systems (e.g., in diagnoses of malfunctioning in chemical or nuclear power plants).

4 Towards (hybrid) extensions

Is it possible to devise purely neural models of other forms of qualitative reasoning? Fragments of first-order logic are natural candidates for an extension, even though the benefits of parallel processing are no longer as evident as in the case of propositional Horn clauses. More importantly, one has to deal with the variable binding problem, and a balanced assessment of whether the various approaches that are now being pursued can pave the way to useful applications in expert systems seems premature, and surely goes beyond the scope of this paper. In this connection, we wish just to recall that the simple weighted-sum, thresholded neuron model described in the previous section is unlikely to provide an adequate technical tool for variable binding: the encoding and passing on of specific variable values by sequences of unitary neuron firings is very inefficient. An alternative, recent approach reported in Ajjanagadde and Shastri (1993) exploits the temporal synchrony of neural activation patterns to represent bindings in limited forms of backward reasoning. Sun and Walz (1991) present a localist neural model for forward chaining on systems of first-order Horn clauses which are constrained by restrictions on the form of their right-hand side. However, the operations that the processing units in this model are to perform are so elaborate that one may well wonder whether the attribute ‘neural’ is still meaningful: a case in point are the processing units representing a given n-place predicate, which are supposed to test whether input real values codify legitimate and mutually consistent bindings for their n variables.

Leaving aside the problem of extensions towards first-order logic, one may still address the problem of neural models of propositional reasoning from incomplete or uncertain information. A naive extension of this sort, starting from the rules described in the previous section, is obtained by modifying the threshold value for a neuron **h** representing the literal on the right-hand side of a rule r with k literals on its left-hand side, each of them capable of firing a unitary impulse on **h**:

$$s_h = k - (\epsilon + \eta) \quad (0 < \epsilon < 1; \eta \geq 1), \qquad (2')$$

Rules implemented in this way fire with data only partially matching their left-hand side. One may also exploit this partial match mechanism in, e.g., qualitative causal reasoning, in order to determine a partial ordering of plausibility between competing explanatory hypotheses, if the amount of excitation received by neurons representing the competing hypotheses in the right-hand side of rules is in various degrees higher than their threshold. To achieve this effect, the neuron **h** of fig. 1 is to be replaced by a “dropper” neuron **d** (fig. 2), which transforms a certain value E of excitation into a number m of consecutive impulses, where m is proportional to E (say, $m \cdot c = E$). $i_1, \ldots, i_n$ are input neurons generating the excitation of **d**, and the dashed line is a negative feedback with value $-c$. Moreover, **d** is endowed with permanent memory, that is, $\delta_d = 1$ is the constant decay function δ_i , and its threshold $\eta < c$ allows **d** to drop all the excitation collected from its inputs under the form of consecutive unitary output impulses.

Fig. 2: Dropper neuron

The input neurons $i_1, \ldots, i_n$ may be interpreted as representing supporting evidence for the hypothesis represented by **d**. Whenever **d** emits an impulse, the negative feedback connection determines a constant value c to be subtracted from the residual excitation value. Thus, **d** keeps on firing until all the excitation E stored in it is dropped away in the form of $m = E/c$ consecutive impulses.

This number m might be used to characterize the degree of plausibility of an hypothesis represented by $\mathbf{d}$, to be compared with the degree of plausibility of other competing hypotheses represented by dropper neurons. Clearly, the permanent memory function plays a crucial role in this mechanism.

The firing of rules under a partial match achieved by means of setting (2') may be a desirable property when rules are used for simulating similarity-based, commonsense reasoning, where loose contextual associations play a central role (see, e.g., Sun (1994)). But these considerations can only very cautiously be generalized to the inherently *brittle* modes of reasoning used in expert systems: principled restrictions on rule firing are required when the *correctness* of diagnoses or classifications is at stake. The best one can do, in our view, to fulfil this desideratum is to have rule firing reflect rigorous models of reasoning under incomplete or uncertain knowledge, so that the uncertainty attached to the conclusions reached by an expert system can be evaluated within a relatively robust conceptual framework.

If one takes seriously this constraint, extensions of the production system neural model of section 4 towards forms of quantitative uncertain reasoning, such as probability or certainty factor propagation, are bound to go hybrid. The main reason is that heavy numerical calculations are required to determine the uncertainty value of the conclusions reached by these modes of inference. In principle, that is, disregarding efficiency constraints, these calculations can be carried out even by suitably organized nets of weighted-sum, thresholded neurons. In practice, the job of subnets devoted to performing these calculations is more efficiently performed by conventional processing units (see Aiello *et al.* (1995b)).

5 Concluding remarks

Our approach to the design of neural modules for rule-based expert systems is characterized by the following choices. First, a localist semantic interpretation is to be prevalently adopted. This is broadly motivated by the requirement of semantic transparency for expert system knowledge representation and inferencing. Second, automatic learning of rules is more suitable, in view of both complexity considerations and the need for adequate justification protocols, for tasks consisting of one-step classification into a small number of classes. Third, rule firing must reflect rigorous models of reasoning under incomplete or uncertain knowledge, so that the uncertainty attached to their conclusions can be properly evaluated. Providing an informative, stepwise justification for the conclusions reached by a neural expert system is jointly facilitated by these constraints. Systems that are designed fulfilling these constraints remain, with the exceptions mentioned above, brittle on the whole, much in the way that traditional symbolic systems are. Brittleness, however, seems to be an almost unavoidable drawback of *reliable*, practically usable expert systems. Thus, in our view, the chief present interest of ANN's in expert systems is the possibility of exploiting the parallel processing potential of neural networks.

References

Aiello A., Burattini E., Tamburrini G. (1995a): "Purely neural, rule-based diagnostic systems. I. Production rules", *Int. J. of Intelligent Systems* **10**, 735-749.

Aiello A., Burattini E., Tamburrini G. (1995b): "Purely neural, rule-based diagnostic systems. II. Uncertain reasoning", *Int. J. of Intelligent Systems* **10**, 751-769.

Ajjanagadde V., Shastri L. (1993): "From simple associations to systematic reasoning", *Behavioral and Brain Sciences* **16**, 417-494.

Badr, A. (1993): "N-tuple classifier for ECG signals", *Proc. of Weightless Neural Network Workshop,* York, UK, 29-32.

Burattini E., De Gregorio M. (1994): "Qualitative abduction and prediction", *Information and Decision Technologies* **19**, 471-481.

Burattini E., Tamburrini G. (1992): "A pseudo-neural system for hypothesis selection", *Int. J. of Intelligent Systems* **7**, 521-545.

Burattini E., Pasconcino A., Tamburrini G. (1995): "A chunking mechanism in a neural system for the parallel processing of propositional production rules", *Mathware and Soft Computing* **2**, 85-116.

De Gregorio, M. (1994): "Experiments with Wisard-like systems in visual data acquisition", Tech. Rep., Istituto di Cibernetica C.N.R., Arco Felice (Napoli).

De Gregorio M. (1996): "Integrating inference and neural classification in a hybrid system for recognition tasks", this issue of *Mathware and Soft Computing.*

Gallant, S.I. (1993): *Neural Network Learning and Expert Systems,* MIT Press, Cambridge, MA.

Gutknecht M., Pfeifer R. (1990): "An approach to integrating expert systems with connectionist networks", *AICOM* **3**, 116-127.

Judd, S. (1990): *Neural Network Design and the Complexity of Learning,* MIT Press, Cambridge, MA.

Puppe, F. (1993): *Systematic Introduction to Expert Systems,* Springer Verlag, Berlin.

Steels, L. (1989): "Connectionist problem solving - An AI Perspective", in R. Pfeifer *et al.* (eds.), *Connectionism in Perspective,* North- Holland, Amsterdam, 215-228.

Sun, R. (1994): *Integrating Rules and Connectionism for Robust Commonsense Reasoning,* John Wiley, New York.

Sun R., Waltz D. (1991): "A neurally inspired massively parallel model of rule-based reasoning", in B. Soucek and the IRIS group (eds.), *Neural and Intelligent Systems Integration,* John Wiley, New York, 341-382.

Towell G.G., Shavlik J.W. (1994): "Knowledge-based Artificial Neural Networks", *Artificial Intelligence* **70**, 119-165.