Analytical approaches to the neural net architecture design*

W J Christmas, J Kittler and M Petrou
Department of Electronic and Electrical Engineering
University of Surrey
Guildford GU2 5XH, United Kingdom

Artificial Neural Networks (ANN) have often been used successfully in problems of classification. Their main drawback is the long training times required particularly for problems of large dimensionality. This is in contrast to the human vision system which is capable of learning from a small number of examples. Another problem with ANNs is the lack of any theoretical guidance concerning the choice of number of nodes, activation functions etc. In this paper we first formulate the problem of object labelling and classification on firm mathematical foundations, within the framework of probabilistic relaxation and then we map it onto the conventional architecture of a multilayer perceptron. This way we can give definite interpretation to the weights, response functions and nodes of the system with specific guidance concerning the choice of the various functions and parameters. Further, we end up with a system which requires very few samples for training, thus emulating the human vision system in that respect.

1. INTRODUCTION

The pace at which pattern classification algorithms have been successfully applied to new problems increased dramatically over the past few years. Much of this activity has been motivated by developments in the field of artificial neural networks (ANN). Good overviews of these algorithms are available in [1–5]. One important characteristic of neural network classifiers is that network outputs provide estimates of a posteriori class probabilities [6] required in a Bayesian minimum error classifier [7]. Indeed, if we view neural networks as a universal methodology for function approximation, then in the case of pattern classification problems the functions to be approximated during training are the a posteriori class probabilities.

The function approximation viewpoint has recently thrown light on empirical behaviour of neural networks [8–11]. The findings have particularly dramatic implications on large scale problems (inputs of high dimensionality). Such problems notably arise in image and vision processing, but even more modest applications raise the question of the practicability of effectively controlling the approximation and estimation errors. The pragmatist's answer to these issues is to resort to heuristics in arbitrarily breaking the classification problem into a multistage process with simple architectures servicing each stage. Such an approach is likely to compromise the optimality of the performance. In any case it

*This work was supported by the Science and Engineering Research Council, UK (GR/J 89255).

does not avoid the requirement for interminable training efforts to reach a reasonable solution and for a multitude of architectural alternatives to be considered in order to gain confidence in the network being able to extract all the relevant information.

These drawbacks are conspicuously in contrast with the functional characteristics of the human nervous system which can learn a pattern from one or a small number of training samples. The learnt model is refined gradually only to cope with pattern class overlaps and ambiguity. The relative importance of stimuli in class definition is rapidly established and presumably reflected in the synaptic strengths.

In a recent work we have developed a Bayesian framework for designing cooperative decision making processes which exploit observational evidence and contextual information relating to objects to be classified [12,13]. This work evolved from earlier results in contextual decision making [14–16]. Most importantly, the pattern classification processes developed have been shown to map on conventional artificial neural network architectures [17,18]. The objective of the work was to establish a link which would facilitate the implementation of the cooperative decision making processes on special purpose hardware neural net architectures that are becoming available. However, the implication of the relationship appears to be reaching far beyond this original goal. In this paper we demonstrate that it can offer an analytical route to designing neural networks as far as the number of nodes and layers, node interconnections, the choice of nonlinearity for the activation function and feedback mechanisms are concerned. An equally important result of the work is that the proposed network training schemes can overcome the theoretical requirements imposed by brute force function approximation methods. The training schemes allow the ANN designer to obtain a rough estimate of the ANN network weights just from one or very few examples of pattern classes to emulate the capability of the human nervous system.

The principal idea of the approach is to represent each pattern to be classified in terms of pattern primitives. The pattern classification process then involves the identification of these primitives. The benefit of introducing pattern primitives is that one can then describe the measurement process model in terms of the conditional distributions rather than the unconditional ones, which should dramatically reduce the order of interactions of the input stimuli. At a first glance this may appear to be possible only at the expense of exponential complexity of interpretation, but it has been shown [12] that this is not the case. The recognition problem complexity is at most of second order polynomial in terms of the number of these primitives and it is believed it can be substantially reduced by means of pruning or by the introduction of dictionaries of admissible label configurations. As a result the probabilistic analysis of the pattern classification problem then allows the appropriate structure of the ANN to be identified, together with the activation function nonlinearities. It also provides guidelines for the initial choice of network weights and a mechanism for their adaptation during training.

The problem formulation involves the specification of pattern primitives, their relations and their respective distributions. The a posterior probability functions of pattern primitive labellings are first developed by means of probability calculus. Thus in contrast to conventional approaches these a posteriori probability functions are not approximated directly. Instead, these functions are expressed in terms of simple components that define and map on a neural net architecture.

We show that it is possible to exploit the relationship between the conditional probability distributions of the relational measurements describing pattern primitives and the neural network weights. The nature of the relationship is first established. The network weights are then determined from the statistical description of these measurement (input stimuli) distributions. The statistical descriptors are obtained during training by means of standard statistical inference techniques.

The paper is organised as follows. First, two basic formulations of the problem of labelling networks of objects are introduced in Section 2. The various contextual decision making schemes that can be developed from the *object centered* formulation are overviewed in Section 3. In Section 4 we outline how object centered labelling schemes map on a multilayer perceptron-like architecture. Finally, Section 5 concludes with a summary of the paper.

2. PROBLEM FORMULATION

Let us consider a set of objects $a_j, j = 1, \ldots, N$ arranged in a network with a particular neighbourhood system.

Each object a_j has an associated measurement vector $\mathbf{x}_j$. Each component of vector $\mathbf{x}_j$ denotes one of three types of measurements:

1. Binary relation measurements $A_{ji}^k,\ k = 1, 2, \ldots, m$ between the j^{th} and i^{th} objects.
2. Unary relation measurements $y_j^l,\ l = 1, 2, \ldots, r$ from which the binary relations are derived.
3. Unary relation measurements $v_j^i,\ i = 1, 2, \ldots, n$ which augment the observational evidence about node j but do not serve as a basis for deriving binary relation measurements A_{ji}^k.

Let us arrange these measurements into vectors as follows:

$$\mathbf{A}_j = \begin{bmatrix} A_{j1} \\ \vdots \\ A_{j(j-1)} \\ A_{j(j+1)} \\ \vdots \\ A_{jN} \end{bmatrix} \tag{1}$$

where $A_{ji} = [A_{ji}^1, \ldots, A_{ji}^m]^T$. For the unary relations we have $\mathbf{y}_j = [y_j^1, \ldots, y_j^r]^T$ and $\mathbf{v}_j = [v_j^1, \ldots, v_j^n]^T$. Thus $\mathbf{x}_j$ is an $[m(N-1) + r + n]$ dimensional vector which can be written as

$$\mathbf{x}_j = \begin{bmatrix} \mathbf{v}_j \\ \mathbf{y}_j \\ \mathbf{A}_j \end{bmatrix} \tag{2}$$

We wish to assign each object a_j a label θ_j. Following the conventional Bayesian approach, object a_i would be assigned to class ω_r based on the information conveyed

by measurement vectors $\mathbf{v}_i$ and $\mathbf{y}_i$ according to the minimum error decision rule [6]. In contrast, here we wish to decide about label θ_i using not only the information contained in unary relation measurements relating to object a_i but also any context conveyed by the network. In other words we wish to utilise also the binary relation measurements, i.e. the full measurement vector $\mathbf{x}_i$ plus all the information about the other objects in the network contained in $\mathbf{x}_j$, $\forall j \neq i$. This is a general statement of the problem but in order to develop contextual labelling schemes our formulation will have to be somewhat more precise.

The first important issue to settle is whether we wish to aim at *object centered* or *message centered* interpretation. In object centered interpretation the emphasis is on one node at a time. Contextual information is used to reduce the ambiguity of labelling a single object. Note that object centered interpretation does not guarantee that the global interpretation makes sense. For example, individually most likely object categories in a character recognition problem will not necessarily combine into valid words. The use of context merely reduces the chance of the global labelling being inconsistent.

In contrast, message centered interpretation is concerned with getting the message conveyed by sensory data right. In our text recognition problem the main objective of message centered labelling would be to label characters so that each line of text gives a sequence of valid words.

The choice between object centered and message centered interpretation will depend in the first instance on the application in hand. For example, if we search for a specified object in an image without requiring to understand the entire content of the image, object centered interpretation might be most appropriate. On the other hand, if global understanding of sensory data is at stake, the labelling task should be posed as a message centered interpretation problem.

Generally speaking, in message centered interpretation we search for a joint labelling $\theta_1 = \omega_{\theta_1}, \theta_2 = \omega_{\theta_2}, \ldots, \theta_N = \omega_{\theta_N}$ which explains observations $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ made on the objects in the network. The most appropriate measure of fit between data and interpretation (but by no means the only one) is the a posteriori probability $P(\theta_1 = \omega_{\theta_1}, \ldots, \theta_N = \omega_{\theta_N} | \mathbf{x}_1, \ldots, \mathbf{x}_N)$. For the sake of brevity we shall denote this probability function as $P(\theta_1, \ldots, \theta_N |\ \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$ whenever this short hand notation does not compromise the clarity of exposition. The Bayesian approach of maximizing a posteriori probability (MAP) of joint labelling which is referred to in the literature as MAP estimation amounts to the following decision rule:

$$\begin{aligned} &\text{assign} \quad \theta_1 \rightarrow \omega_{\theta_1}, \ldots, \theta_N \rightarrow \omega_{\theta_N} \quad \text{if} \\ &P(\theta_1 = \omega_{\theta_1}, \ldots, \theta_N = \omega_{\theta_N} | \mathbf{x}_1, \ldots, \mathbf{x}_N) = max_{\Omega_1, \ldots, \Omega_N} P(\theta_1, \ldots, \theta_N | \mathbf{x}_1, \ldots, \mathbf{x}_N) \end{aligned} \tag{3}$$

where Ω_i is the set of labels admitted by object a_i. For simplicity we shall assume that $\forall i$, $\Omega_i = \{\omega_0, \omega_1, \ldots, \omega_M\} = \Omega$, where ω_0 is the null label used to label objects for which no other label is appropriate.

The object centered counterpart computes instead $P(\theta_i = \omega_{\theta_i} | \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$, the a posteriori probability of label θ_i given all the observations. The main difference between message and object centered interpretation can be brought out by expressing both proba-

bilities using the Bayes formula. Starting with the a posteriori probability of joint labelling

$$P(\theta_1, \ldots, \theta_N | \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) = \frac{p(\mathbf{x}_1, \ldots, \mathbf{x}_N | \theta_1, \ldots, \theta_N) P(\theta_1, \ldots, \theta_N)}{p(\mathbf{x}_1, \ldots, \mathbf{x}_N)} \tag{4}$$

we note that for given observations the joint probability density function value in the denominator is fixed and therefore the left hand side is proportional to the product in the numerator. The first term of the product, the conditional joint probability density function of measurement vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$ models the measurement process whereas the second term embodies our a priori knowledge of the likelihood of various combinations of labels occurring. It is our global, world model.

For the object centered labelling we can write

$$P(\theta_i = \omega | \mathbf{x}_1, \ldots, \mathbf{x}_N) = \frac{p(\mathbf{x}_1, \ldots, \mathbf{x}_N | \theta_i = \omega) P(\theta_i = \omega)}{p(\mathbf{x}_1, \ldots, \mathbf{x}_N)} \tag{5}$$

where $P(\theta_i = \omega)$ is the a priori probability of label θ_i taking value ω. Again the denominator in (5) can be dismissed. Expanding the first term of the numerator over all possible labellings in the usual fashion, i.e.

$$\begin{aligned} & p(\mathbf{x}_1, \ldots, \mathbf{x}_N | \theta_i = \omega) = \\ & \quad = \sum_{\Omega_1} \ldots \sum_{\Omega_{i-1}} \sum_{\Omega_{i+1}} \ldots \sum_{\Omega_N} p(\mathbf{x}_1, \ldots, \mathbf{x}_N, \theta_1, \ldots \theta_{i-1}, \theta_{i+1} \ldots, \theta_N | \theta_i = \omega) \\ & = \sum_{\Omega_1} \ldots \sum_{\Omega_{i-1}} \sum_{\Omega_{i+1}} \ldots \sum_{\Omega_N} p(\mathbf{x}_1, \ldots, \mathbf{x}_N | \theta_1, \ldots \theta_i = \omega \ldots, \theta_N) P(\theta_1, \ldots \theta_{i-1}, \theta_{i+1} \ldots, \theta_N | \theta_i = \omega) \end{aligned} \tag{6}$$

we find

$$\begin{aligned} & P(\theta_i = \omega | \mathbf{x}_1, \ldots, \mathbf{x}_N) = \\ & \quad = \frac{\sum_{\Omega_1} \ldots \sum_{\Omega_N} p(\mathbf{x}_1, \ldots, \mathbf{x}_N | \theta_1, \ldots \theta_i = \omega \ldots, \theta_N) P(\theta_1, \ldots \theta_i = \omega \ldots, \theta_N)}{p(\mathbf{x}_1, \ldots, \mathbf{x}_N)} \end{aligned} \tag{7}$$

Thus computing the probability of a particular label ω on a single object a_i amounts to scanning through all the possible combinations of labels $\theta_1, \ldots, \theta_N$ with label θ_i set to ω and summing up the corresponding products of the respective joint measurement and label probabilities.

Inspecting the expressions for $P(\theta_1, \ldots, \theta_N | \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$ and $P(\theta_i = \omega | \mathbf{x}_1, \ldots, \mathbf{x}_N)$ in (4) and (7) respectively reveals that they are both defined in terms of the same ingredients, namely the a priori probability distribution of joint labelling, $P(\theta_1, \ldots, \theta_N)$ and the conditional probability density function $p(\mathbf{x}_1, \ldots, \mathbf{x}_N | \theta_1, \ldots, \theta_N)$. It is apparent that for any practical value of N and M it will not be feasible to apply techniques of statistical inference to estimate these probability distributions. However, in many practical situations simplifying assumptions can be made to make the computation of the a posteriori probabilities feasible. In particular, an important and physically realistic assumption regarding the unary measurement process distribution is that the outcomes of measurements are conditionally independent.

$$p(\mathbf{v}_1, \mathbf{y}_1 \ldots, \mathbf{v}_N, \mathbf{y}_N | \theta_1, \ldots \theta_i, \ldots, \theta_N) = \prod_{i=1}^{N} p(\mathbf{v}_i, \mathbf{y}_i | \theta_i = \omega_{\theta_i}) \tag{8}$$

Also, for binary relations we assume that

$$p(A_{i1},\ldots,A_{iN}|\theta_1,\ldots\theta_i,\ldots,\theta_N)=\prod_{j\neq i}p(A_{ij}|\theta_i,\theta_j) \tag{9}$$

Finally, the idea of a dictionary model or a Markov random field model [12–15] can be used to simplify the prior probability of joint labelling $P(\theta_1,\ldots\theta_i=\omega\ldots,\theta_N)$.

3. PROBABILISTIC RELAXATION

Under some mild conditional independence assumptions concerning measurements $\mathbf{x}_j$, $\mathbf{y}_j$ and A_{ij}, $\forall j$ the object centered labelling formulation (7) leads to an iterative probability updating formula [12]:

$$P^{(n+1)}\left(\theta_i\leftarrow\omega_{\tau_i}\right)=\frac{P^{(n)}\left(\theta_i\leftarrow\omega_{\tau_i}\right)Q^{(n)}\left(\theta_i\leftarrow\omega_{\tau_i}\right)}{\sum_{\omega_\lambda\in\Omega}P^{(n)}\left(\theta_i\leftarrow\omega_\lambda\right)Q^{(n)}\left(\theta_i\leftarrow\omega_\lambda\right)} \tag{10}$$

where $P^{(n)}(\theta_i\leftarrow\omega_)$ denotes the probability of label ω_{θ_i} at object a_i at the n^{th} iteration of the updating process and the quantity $Q^{(n)}(\theta_i\leftarrow\omega_)$ expresses the support the label $\theta_i\leftarrow\omega_\alpha$ receives at the n^{th} iteration step from the other objects in the scene, taking into consideration the binary relations that exist between them and object a_i. After the first iteration (n=1) the computed entity is the contextual a posteriori class probability $P(\theta_i=\omega_{\theta_i}|\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N)$. As n increases, the updating scheme drives the probabilistic labelling into a hard labelling.

The support $Q^{(n)}(\theta_i=\omega_{\theta_i})$ is defined as

$$\begin{aligned}&Q^{(n)}(\theta_i=\omega_{\theta_i})=\\&=\sum_{\omega_{\theta_j},j\in N_i}\frac{1}{\hat{p}\left(\theta_i=\omega_{\theta_i}\right)}\left\{\prod_{j\in N_i}\frac{P^{(n)}\left(\theta_j=\omega_{\theta_j}\right)p\left(A_{ij}|\theta_i=\omega_{\theta_i},\theta_j=\omega_{\theta_j}\right)}{\hat{p}\left(\theta_j=\omega_{\theta_j}\right)}\right\}\times\\&\quad\times P\left(\theta_j=\omega_{\theta_j},\forall j\in N_i\right)\end{aligned} \tag{11}$$

where $p\left(A_{ij}|\theta_i=\omega_{\theta_i},\theta_j=\omega_{\theta_j}\right)$ is the compatibility coefficient quantifying the mutual support of the labelling $(\theta_i=\omega_{\theta_i},\theta_j=\omega_{\theta_j})$. N_i denotes the index set of all nodes excluding the node i, i.e. $N_i=\{1,2,\ldots,i-1,i+1,\ldots,N\}$. It is worth noting that, when binary relations are not used, the support function (11) becomes the standard evidence combining formula developed in [14], i.e.

$$\begin{aligned}&Q^{(n)}(\theta_i=\omega_{\theta_i})=\\&=\sum_{\omega_{\theta_j},j\in N_i}\frac{1}{\hat{p}\left(\theta_i=\omega_{\theta_i}\right)}\left\{\prod_{j\in N_i}\frac{P^{(n)}\left(\theta_j=\omega_{\theta_j}\right)}{\hat{p}\left(\theta_j=\omega_{\theta_j}\right)}\right\}\times P\left(\theta_j=\omega_{\theta_j},\forall j\in N_i\right)\end{aligned} \tag{12}$$

On the other hand, when no additional unary relation measurements are available apart from the set used for generating the binary measurements, the support reduces to

$$\begin{aligned}&Q^{(n)}(\theta_i=\omega_{\theta_i})=\\&=\sum_{\omega_{\theta_j},j\in N_i}\frac{1}{\hat{p}\left(\theta_i=\omega_{\theta_i}\right)}\left\{\prod_{j\in N_i}p\left(A_{ij}|\theta_i=\omega_{\theta_i},\theta_j=\omega_{\theta_j}\right)\right\}P\left(\theta_j=\omega_{\theta_j},\forall j\in N_i\right)\end{aligned} \tag{13}$$

The probability updating rule (10) in this particular case will act as an inefficient maximum value selection operator. Thus the updating process can be terminated after the first iteration, the maximum contextual a posteriori label probability selected and set to unity while the probabilities of all the other labels are set to zero.

The support function (11) exhibits exponential complexity. In practice its use, depending on application, could be limited only to a contextual neighbourhood in the vicinity of the object being interpreted. Such a measure is appropriate for instance in the case of edge and line postprocessing, where the objects to be labelled are pixel sites. A small neighbourhood, say a 3 by 3 window may be sufficient to provide the necessary contextual information. In any case, by iteratively updating the pixel label probabilities using formula (10) contextual information would be drawn from increasingly larger neighbourhoods of each pixel.

A more dramatic, complementary reduction in the computational complexity is achieved by noting that in practice many potential label configurations in the contextual neighbourhood of an object are physically inadmissible. By listing the admissible labellings in a dictionary, the above support function can be evaluated by summing up only over the entries $\left(\theta_j = \omega_{\theta_j}^k, \forall j \in N_i\right)$, $\forall k$ in the dictionary, i.e.

$$
\begin{aligned}
&Q^{(n)}(\theta_i = \omega_{\theta_i}) = \\
&= \sum_{k=1}^{Z(\omega_{\theta_i})} \frac{1}{\hat{p}\left(\theta_i = \omega_{\theta_i}\right)} \left\{ \prod_{j \in N_i} \frac{P^{(n)}\left(\theta_j = \omega_{\theta_j}^k\right) p\left(A_{ij} | \theta_i = \omega_{\theta_i}, \theta_j = \omega_{\theta_j}^k\right)}{\hat{p}\left(\theta_j = \omega_{\theta_j}^k\right)} \right\} \times \\
&\quad \times P\left(\theta_j = \omega_{\theta_j}^k, \forall j \in N_i\right)
\end{aligned} \tag{14}
$$

where $Z(\omega_{\theta_i})$ denotes the number of dictionary entries with label θ_i set to ω_{θ_i}.

In many labelling problems neither of the above simplifications of the support function is appropriate. For instance, in correspondence matching tasks or object recognition all features of an object interact directly with each other. Moreover, without measurements, no labelling configuration is a priori more likely than any other. Then it is reasonable to assume that the prior probability of a joint labelling configuration can be expressed as

$$
P\left(\theta_j = \omega_{\theta_j}, \forall j \in N_i\right) = \prod_{j \in N_i} \hat{p}\left(\theta_j = \omega_{\theta_j}\right) \tag{15}
$$

Substituting (15) into (11) and noting that each factor in the product in the above expression depends on the label of only one other object apart from the object a_i under consideration, we can simplify the support computation as

$$
Q^{(n)}(\theta_i \leftarrow \omega_\alpha) = \prod_{j \in N_i} \sum_{\omega_\beta \in \Omega} P^{(n)}\left(\theta_j \leftarrow \omega_\beta\right) p\left(A_{ij} \mid \theta_i \leftarrow \omega_\alpha, \theta_j \leftarrow \omega_\beta\right) \tag{16}
$$

It is interesting to note that through this simplification the exponential complexity of the problem is eliminated.

The iteration scheme can be initialised by considering as $P^{(0)}(\theta_i \leftarrow \omega_{\tau_i})$ the probabilities computed by using the unary attributes only, *i.e.*

$$
P^{(0)}(\theta_i \leftarrow \omega_{\tau_i}) = P(\theta_i \leftarrow \omega_{\tau_i} | \mathbf{v}_i, \mathbf{y}_i) \tag{17}
$$

We discuss this initialisation process in detail elsewhere [12].

4. NEURAL NET IMPLEMENTATION

The aim of this section is to demonstrate how the cooperative processes discussed in Section 3 can be mapped on a neural net architecture. The purpose of the mapping is at least twofold. First of all, neural net computation has inspired and motivated considerable activity in specialist hardware and software architecture design and implementation. Software, and hardware systems, including VLSI chips, have been developed to implement various families of neural networks. A successful mapping of contextual decision making processes on such systems would facilitate their wide applicability.

The second, and perhaps more significant purpose is to argue that the mapping process could offer a route to neural network design which is not plagued by the typical problems associated with the development of neural network solutions to pattern classification tasks: lack of guidelines for the choice of architecture and node connectivity, lack of data, unacceptably long training phase, and last but not least, the lack of criteria for the selection of the node activation functions and for the weight initialization.

Rather than attempting a comprehensive coverage of all the cooperative processes discussed in the paper, we shall illustrate the basic ideas on a specific contextual labelling algorithm. In particular, we shall consider the probabilistic relaxation scheme in (10) with the support function given by formula (16). The neural network performing the same computation is presented in Figure 1.

It is basically a multilayer perceptron with two main layers and an auxiliary layer which performs a normalisation computation to maintain the network outputs in the zero - one range and ensuring that the outputs representing label excitation for each object primitive sum up to unity. There are only N such auxiliary units as there are N object primitives to be labelled. The inputs P_{ij} to the multilayer perceptron are computed by a noncontextual artificial neural network which at its input is stimulated by unary relation measurements observed for each object primitive. We shall not dwell on the methodology that can be used for designing such a neural network as the number of unary measurements one deals with is normally relatively small and therefore the neural net design problems identified earlier are not applicable. The outputs P_{ij} of this initializing network correspond to the a posteriori (noncontextual) label probabilities for the object primitives based on the unary relations.

The main network has the normal characteristics of a typical multilayer perceptron design. The number of units in the second layer expands whereas for the final layer it contracts. In our design the number of output units is the same as the number of input units to the network with a separate unit for each object primitive/label combination.

The weights applied to the links between the input nodes and the nodes of the second layer are defined by the binary relation probabilities which can be easily estimated during training by techniques of statistical inference rather than weight adaptation. The output of each node q_i^{jk} represents the support for label k on object primitive j from object primitive i. Note that the activation function of the units in the second layer is the *log* function and not the usual sigmoid. The connections from the second layer to the third layer have no weights associated with them. The activation function of the units in this third layer is the exponential function.

A distinguishing characteristic of the network is the feedback link from the normalised

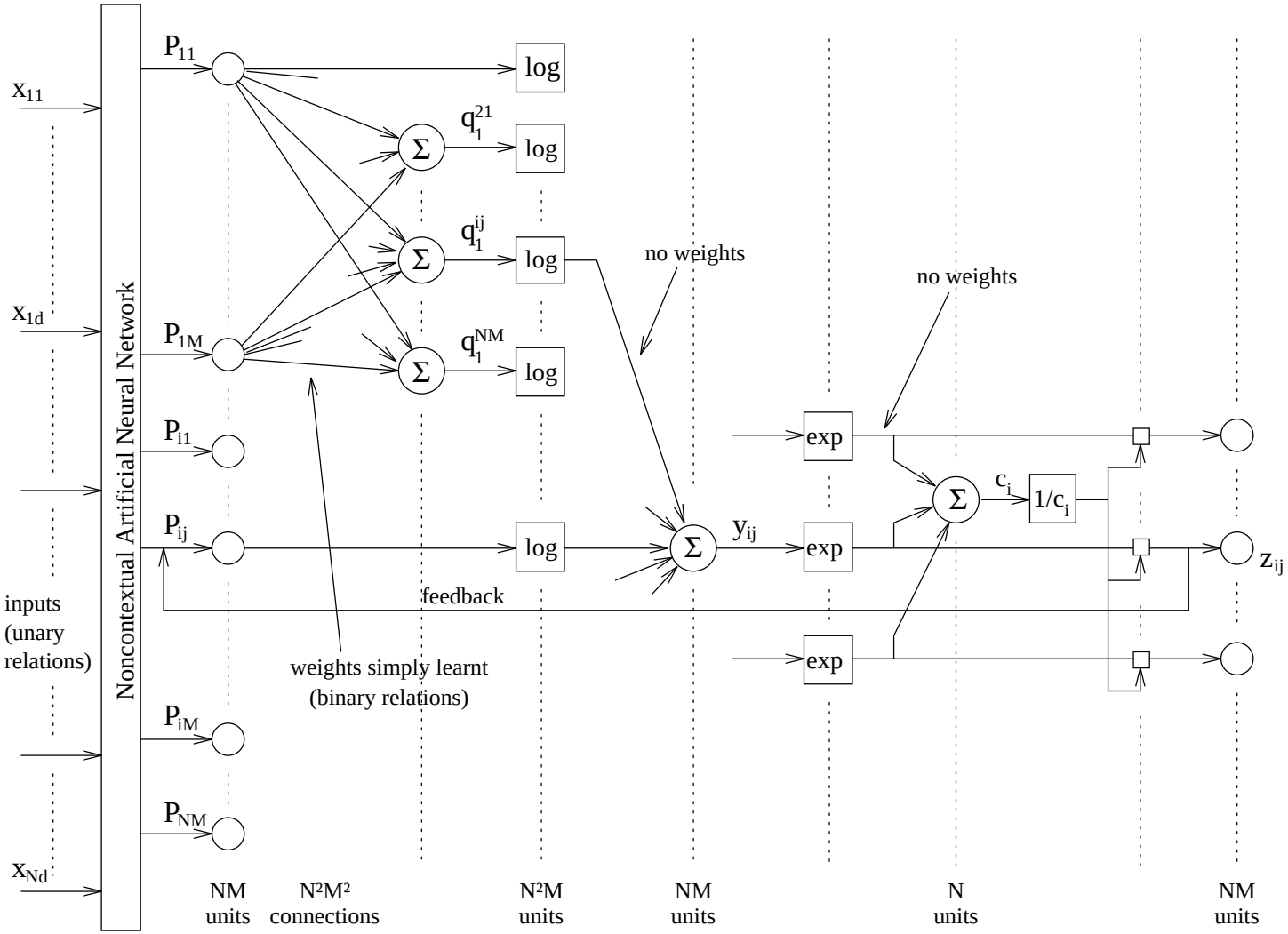

Figure 1: Multilayer perceptron implementation of the probabilistic relaxation scheme with the support function given by equation (16)

output to the input. If this link is broken, the output units generate the contextual probabilities of object primitive labelling corresponding to a single iteration of the label updating process in (10). The feedback forces one of the outputs associated with each object primitive to unity and all the others to zero.

We have thus demonstrated that a neural network for a complex pattern classification problem can be designed by analysing the nature of the problem first. The analysis makes it possible to incorporate realistic assumptions into the solution of the problem. This in turn often facilitates a dramatic reduction in complexity of the classification problem. Thus instead of having to approximate a posteriori probability functions in high dimensional observation spaces for which a massive architecture of unknown node connectivity would be required, one can express these a posteriori probabilities in terms of functions of simple components. This explicitly specifies the required architecture of the neural network, node connectivity and the functional form of the activation functions. But even more importantly, the training of such a network becomes a simple task of inferring the probability distributions of the relevant measurements in low dimensional spaces. A first guess of these distributions can easily be made from the presentation of very few prototypical patterns to the system. In this sense the training capability of the proposed approach emulates closely that of the human central nervous system which also can learn complex patterns very efficiently from just a few experiences.

5. CONCLUSIONS

In the paper, it was demonstrated that probabilistic relaxation labelling processes can be mapped onto a neural net architecture, in particular the multilayer perceptron. This has the important implication that cooperative decision making schemes can provide an approach to designing artificial neural networks with the benefit of simple training, and of the ANN architecture and activation functions being uniquely specified. The design approach appears to offer an attractive alternative to the conventional ANN design techniques.

REFERENCES

1. R P Lippmann, An introduction to computing with neural nets, *IEEE ASSP Magazine*, **4**, 4-22, 1987.
2. D R Hush and B G Horne, Progress in supervised neural networks, *IEEE Signal Processing Magazine*, **10**, 8-39, 1993.
3. G A Carpenter and S Grossberg, A self-organizing neural network for supervised learning, recognition and prediction, *IEEE Communications Magazine*, 38-49, 1992.
4. F Fogelman Soulie, B Lamy and E Viennet, Multi-modular neural network architectures for pattern recognition: Applications in optical character recognition and human face recognition, *Int. Jnl of Pattern Recognition and Artificial Intelligence*, 1993.
5. V Cherkasky and H Wechsler, From statistics to neural networks, Springer Verlag, Berlin, 1994.
6. P A Devijver and J Kittler, Pattern recognition: A statistical approach, Prentice-Hall, Englewood Cliffs, NJ, 1982.

7. M D Richard and R P Lippmann, Neural network classifiers estimate Bayesian aposteriori probabilities, *Neural Computing*, **3**, 461-483, 1992.
8. J E Moody, The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems, in *Advances in Neural Information Processing Systems 4*, J E Moody, S J Hanson and R P Lippmann, eds., Morgan Kaufmann, San Mateo, CA, 1992.
9. J Friedman, Multivariate adaptive regression splines, *Annals of Statistics*, **19**, 1-141, 1991.
10. H White, K Hornik, M Stinchcombe, Multilayer feedforward networks as universal approximators, in *Artificial Neural Networks*, H White, eds., 13-28, Blackwell, 1992.
11. A R Barron, Approximation and estimation bounds for artificial neural networks, in *Proc 4th Workshop on Computational Learning Theory*, 243-249, 1992.
12. J Kittler, W J Christmas and M Petrou, Probabilistic relaxation for matching problems in computer vision, *Proc 4th Intern. Conference on Computer Vision*, 666-673, Berlin, 1993.
13. J Kittler, P Papachristou and M Petrou, Combining evidence in dictionary based probabilistic relaxation *Proc 8th Scandinavian Conference on Image Analysis*, 785-793, Tromso, 1993.
14. J Kittler and E R Hancock, Combining evidence in probabilistic relaxation, *Intern. Journal of Pattern Recognition and Artificial Intelligence*, **3**, 29-51, 1989.
15. E R Hancock and J Kittler, Edge labeling using dictionary-based relaxation, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **PAMI-12**, 165-181, 1990.
16. E R Hancock and J Kittler, Discrete relaxation, *Pattern Recognition*, **23**, 711-733, 1990.
17. J Kittler, Relaxation methods and their neural net implementation, in *From statistics to neural networks*, V Cherkasky and H Wechsler, Eds., Springer-Verlag, Berlin, 1994.
18. E R Hancock and J Kittler, An improved error criterion for neural networks, *Proc 7th Scandinavian Conference on Image Analysis*, 1094-1101, Aalborg, Denmark, 1991.