INTERNAL CONFLICTS IN NEURAL NETWORKS

S. Ye. Gilev, A. N. Gorban, Ye. M. Mirkes

Krasnoyarsk AMSE Center, Computing Center,
Krasnoyarsk, 660036, Russia

Hierarchical neural networks consisting of small expert-networks are considered. Algorithms of fast parallel learning are proposed.

1. **Basic structures.** The ideas applied are essentially those put forward in fundamental works [1-3], preceding the neurocomputing "outbreak". They are, first of all, the idea of "Cora" (Cortex) by Bongard and Marr's codons.

The idea of a system of experts: several networks are trained to solve some task. In the process of training the "fields of competence" are captured, with the experts learning to give exact answers in their own field and make not very coarse errors in others. In functioning the experts' voting is processed either with a special device or a program or a specially trained network.

This section describes basic structures in the general form. We'll start with the problem of teaching one expert-network.

Adaptive matrix of signal reception. The network assigns input neurons (in particular, every neuron can be input). The input signals can be fed onto the input neurons through the adaptive accumulator. Introduce denotations. Let

$i=1,\ldots,n$ be the input neuron index,

$j=1,\ldots,k$ be the input signal index;

a_i be the value fed at the input of the i-th input neuron,

$a=(a_i)$ be the input vector,

x_j be the value of the j-th signal,

$x=(x_j)$ be the signal vector;

$M=(m_{ij})$ be the incidence matrix: $m_{ij}=1$, if x_j is fed onto the i-th input accumulator and $m_{ij}=0$ otherwise;

$A=(\alpha_{ij})$ be the adaptive matrix of signal reception:

$$a_i=\sum_{j=1}^{k}\alpha_{ij}x_j \quad (a=Ax). \qquad \text{(EQ1)}$$

H be the estimation function.

The connection between A and M is: if $m_{ij}=0$, then $\alpha_{ij}=0$. In the course of learning A can change in any of the known algorithms, M as a result of special contrasting procedure [4].

Contrasting a reception matrix. The matrix A is contrasted to separate the most significant parameters and force the others turn into zero. Training is divided into three phases: primary learning - contrasting - adjusting.

In the course of primary learning the indice of sensitivity $\varkappa_{ij}$ for each α_{ij} (provided $m_{ij}\neq 0$) are determined. They show how much the estimate of the network operation depends of this parameters.

When contrasting in columns A, each signal x_j is allotted q numbers of neurons $i=i_1(j),\ldots,i_q(j)$ with the greatest $\varkappa_{ij}$. Assume $m_{ij}=1$ at $i=i_1(j),\ldots,i_q(j)$, and $m_{ij}=0$ otherwise.

In analogy, when contrasting in rows each neuron is allotted r signals fed onto it.

Finally, in general contrasting, allotted are s most significant (with the highest $\varkappa$) signal-neuron connections.

The rest turn into zero.

The numbers q, r, s are the parameters of the procedures.

Note. We pay immediate attention to individual procedures, assuming the parameters of the procedures and the final assembly of the algorithms to be different for different tasks. This is in agreement with the structure of the "Neurodesigner" project within the limit of which we carried out the work.

Sensitivity indice $æ_{i,j}$ are first formed for one run of each training pattern by the network. Let l be the number of the training patterns, x^l and a^l *the respective vectors,* $æ^l_{i,j}$ be the sensitivity index for $\alpha_{i,j}$ when solving the given pattern. This is a product $\alpha_{i,j}x^l_j$ by the significance index of the i-th input neuron $æ^l_i$:

$$æ^l_{i,j}=\alpha_{i,j}x^l_j æ^l_i \qquad \text{(EQ2)}$$

If computation of the derivatives of H_l is possible, take

$$æ^l_i=\left|\frac{\partial H_l}{\partial a_i}\right|,\ æ^l_{i,j}=\alpha_{i,j}x^l_j\frac{\partial H_l}{\partial a_i} \qquad \text{(EQ3)}$$

This $æ^l_{i,j}$ is the modulus of the derivative of H_l *in* $m_{i,j}$ formally introduced in (1):

$$a_i=\sum_j \alpha_{i,j}m_{i,j}x_j \qquad \text{(EQ4)}$$

In other cases $æ^l_{i,j}$ can be defined as a sum of the connection weights getting from the i-th neuron (independent of l), a sum of signals, getting from the i-th neuron onto the network neurons, etc.

The indice $æ^l_{i,j}$ for one event of solution of one pattern available, we should pass to a set of patterns and sequence of their solutions (each pattern can be solved several times).

Successive training of experts. When training the next expert the requirements to its functioning on the patterns, which are solved by the previous experts correctly, are reduced - the main thing is not to make on these patterns too coarse errors.

The parallel training of experts. In this case the experts are trained with the "field of competence" in the taskbook gradually separating. At this, the experts must not make too coarse errors in others' areas, while in its own - solve the problem quite precisly.

2. Example of implementation. Consider the problem of pattern recognition. Denote the number of patterns by Q. Each expert has Q output neurons. It votes for the pattern whith the number z ($1 \leqslant i \leqslant Q$), if the z-th output signal for it y_z exeeds the others y_i ($1 \leqslant i \leqslant Q$, $i \neq z$). Let further l be the number of the case, $z(l)$ be the number of the respective pattern, N be the number of a small expert, y^{Nl} be the output signal of the N-th expert when solving the l-th case. The level of reliability h is the required excess of the output signal whith the appropriate number y_z over the other y_i.

Evaluation function is constructed as follows. Denote by e_z the Q-dimensional vector with the z-th coordinate - 1 and the other - 0, $Y_z=\{y \mid y \geqslant y_i; i=1,\ldots,Q; i \neq z\}$. Let the training case be from the class with the number z. Then for this pattern

$$H_h^z(y)=dist(y-he_z, Y_z), \qquad (EQ5)$$

where *dist* is the distance from the point to the set.

The function of penalty for the "too confident incorrect answer" is constructed in analogy:

$$P_h^z(y)=dist(y-he_z, Y_z), \qquad (EQ6)$$

Each expert is learning, minimizing its function of evaluation V_N:

$$V_N=\sum_l w P_{Nl}+w_{Nl}H_N, \qquad (EQ7)$$

where w, $w_{Nl}>0$ are the the weights, $0 \leqslant \varepsilon \leqslant 1$, $P_{Nl}=P_{\varepsilon h}^{z(l)}(y^{Nl})$, $H_{nl}=H_h^{z(l)}(y^{Nl})$.

Learning algorithm and the network structure. The experts are full-connected or layered networks with smooth sigmoidal input-output characteristic, linear accumulators at the inputs and linear connections. The connection coefficients in learning are maintained in the range [-1, 1], we choose the characteristic in the form $f(a)=a/(c+a)$, c is the constant (usually $c=0.1 \div 0.5$).

Evaluation function (7) is minimized in single-step quasi-Newton method (BFGS-formula). The derivatives are computed by the duality principle [4-7].

Redistribution of the weights between the patterns and the experts is constructed as follows: the weights shift from the well solved cases to the badly solved ones and from the experts solving the given

cases badly to those, which solve it well. Give a simplest network algorithm of such a distribution. For the weights we construct a differential equation:

$$\dot{w}= \sum_{M,q} (K_{Mq,N1} w_{Mq} - K_{N1,Mq} w_{N1}) \quad \text{(EQ8)}$$

At the initial moment of learning all $w_{N1}=1$.

The constants of the weight flow rate are determined as follows:

$$K_{N1,Mq}=\begin{cases} AH_{N1}, & \text{if } 1=q \\ BH_q, & \text{if } 1\neq q \end{cases} \quad \text{(EQ9)}$$

where $H_q=\min H_{Mq}$; A, B – are the positive constants.

To make the experts competitive (a strong expert forces a weak one from the task book in general and not for the given task only), it is sufficient to the coefficients (9) to add

$$K_{N1,Mq}=C\sum_1 H_{N1}, \quad \text{(EQ10)}$$

where C is the positive constant.

Equation (8) are easily realized by an other network.

After each optimization step the flow coefficients change, new weights are assigned in compliance with (8) and again – an optimization step. The constants A, B, C specify the procedure of weight selection. At this the time of network functioning can be considered to be fixed,

Annealing. In the course of learning h is gradually increasing, ε – decreasing and w necessarily grows high (from 1 to 10^2–10^3) so, that in the result of learning not a single expert would do too coarse errors, if possible.

The experts' voting is performed according to the following principle: voting is the most confident expert (the one with the maximum output signal exceeding the closest in the value by the greatest number), or voting is processed by a special neuron network, which is trained after completing the basic learning process.

3. Test tasks

Election of the american president [8] – by 12 input signals, describing the political situation in the USA on the eve of the election, 2-neuron fully connected network after 2 steps of signal

exchange between the neurons predicts which party is to win.

Recognizing of succession shift (a standard test problem). The network input is fed a succession of 8 zeros or ones and the result of its cyclic shift one position left or right. Two 3-neuron experts solve in two steps of signal exchange in what direction is the succession moved. For this purpose Boltzmann machine uses 252 neurons [9]. And one 4-neuron network mastered this problem on the whole. At this it pointed out 4 patterns set up erroneously as unsolvable (corrected the typesettings errors).

Among the problems solved were recognition of visual images and prediction of chemical elements' properties. The approach proposed greatly enlarges the information capacity of the network and accelerates learning, but the question how it can help in the progress of understanding the work of the real neuron networks remains to be solved.

The authors are thankful to V. L. Dunin-Barkovsky for useful discussion of the work.

4. References

1. Bongard M. M. "The Recognition Problem". Moscow: Nauka, 1988 (in Russian).
2. Marr D. "A Theory for cerebral Neocortex", Proc. Roy. Soc. Lond. 1970. V. 176. B, p. 161-234.
3. Dunin-Barkovski V. L. "Information processes in neural structures". -Moscow: Nauka, 1978. - 160 pp. (in Russian).
4. Gorban A. N. "Training Neural Networks" . - Moscow: USSR-USA JV ParaGraph, 1990. - 160pp. (in Russian).
5. Bartsev S. I., Okhonin V. A. "Information processing adapttive networks". -Krasnoyarsk: Institute of Physics 1986. - 20pp. (Prepr. / Institute of Physics, USSR, Acad. Sci., Sib. Branch, No 59Б) (in Russian).
6. Bartsev S. I., Gilev S. E., Okhonin V. A. "Duality principle in organization of information processing adaptive networks". Dynamics of chemical and biological systems. Novosibirsk: Nauka, 1989, p. 6-55 (in Russian).
7. Gilev C. E., Gorban A. N., Mirkes E. M. "Several mehods

for accelerating the training process of neural networks in pattern recognition". -Krasnoyarsk: Institute of Biophysics 1990. - 16pp. (Prepr. / Institute of Biophysics, USSR, Acad. Sci., Sib. Branch, No 146Б).

8. Waxman Cory. "Neurocomputers in the Human Sciences. Program: Prediction of US Presidential Elections". -Krasnoyarsk: Institute of Biophysics 1990. - 15pp. (Prepr. / Institute of Biophysics, USSR Acad. Sci., Sib. Branch, No 147Б).

9. A. N. Gorban, Neurocomputing in Siberia, Adv. Model. & Analysis, B, 1992, V. 24, No 2, pp. 21-28.

10. S. Ye. Gilev, A. N. Gorban, Ye. M. Mirkes, Small Experts and internal Conflicts in learnable Neural Networks, Adv. Model. & Analysis, B, 1992, V. 24, No 1, p. 45-50. (see also Doklady Academii Nauk SSSR, 1991, V. 320, No 1, p. 220-223).