ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 111-116

MLP MODULAR NETWORKS FOR MULTI-CLASS RECOGNITION

Philippe SEBIRE and Bernadette DORIZZI

Institut National des Télécommunications
9, rue Charles Fourier
F-91011 Evry Cedex, France
nad@etna.int-evry.fr

Abstract
We present a connectionist modular approach which is potentially able to deal with real-size applications as its size does not increase drastically with the size of the problem. It relies on very simple cooperation schemes of modular MLP networks especially designed for some sub-tasks. Several cutting up are tested : from two or three nets to one network per class. These approaches are compared on a multi-class classification task (recognition of typographic characters) in terms of performance rates.

1. Introduction

Multi Layer Perceptron (MLP) networks have been extensively studied during the last years. In many applications (speech or handwritten character recognition, time series prediction etc...) they have proved to be very efficient classifiers, especially because they do not necessitate a lot of preprocessings. Most often, these studies have been conducted on small size problems with a limited number of classes, which presents the advantage of maintaining the simulation time reasonable. But even in this frame, it has been shown that it is only at the price of a big training set that the results of the digit recognition problem are equivalent to those obtained by the best statistical method [IDA92]. This implies large training times and thus difficulties to optimaly adjust the different parameters of the net. This is even worse when one consider a problem with many classes (more than 50) as the number of training elements and thus the size of the net increase. Moreover, a realistic application is submitted to changes in its specifications (for example, a new class is added, some new examples are considered etc...). So, as no incremental learning is possible with the standard MLP due to its fixed architecture and its mode of functioning, it does not seem to be a very practical tool in real size applications.

A general idea to tackle this problem is to split the global task (N classes) into p subtasks corresponding to less classes. A network is thus designed to solve each subtask and a cooperation module is created to collect the results from these different networks. The splitting of the base may be a-priori [WAI89, BOL91] or a result of the learning phase [JAC91]. In our work we have tested several cutting up of the base: from 3 or 4 nets to one net per class. The cooperation scheme is very simple (logic rule or perceptron at this output). We compare the different results on a multifont character recognition problem of 49 classes (digits, upper and lower-case letters). The characters are first typed, then scanned and segmented. Some of them are

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 111-116

slightly slanted. The input to the different networks is the pixel image that we normalize to a 16x32 matrix with values in {-1,0,1} to have more information about the borders of the characters. This typographic multifont data base is divided into two sub-bases : a training set of 3779 characters and a test set of 3888 characters.

In all our simulations, we have used networks trained with the backpropagation algorithm. In this case, the classification decision is not given by the network, it has to be fixed by the user. The first rule, which is the simplest, consists of declaring the class with the largest value (named class-max) the winner. But it may happen that no class is sufficiently active and, in this case, it is not informative to choose the class-max. We thus define a second decision criterion, which uses two thresholds. Namely a character will be rejected if the largest output value is below a reject threshold. Among the other characters, it could also occur that the responses of the network for two different classes be very close to one another. In this case, we declare that the character is ambiguous (by an ambiguity threshold). The other ones are either recognized or confused. The quality of a network can thus be evaluated with respect to the stability of these 4 labels when the thresholds vary. Saturating networks (for which the introduction of a reject threshold does not degrade the recognition rate too much) are particularly faithful. Corresponding to these different thresholds, we can compute confusion and ambiguity matrices.

2. A network for the global task

We first built a global neural network (module n°1) which is in charge of the classification task. Its performances will be a reference for the following modular architectures. We use a structure introduced by LeCun [LEC90] for handwritten digit recognition, with local and shared-weight connections.
This structure (see figure 1), which may look complicated at first, is very efficient in solving this task. Indeed, there is a problem in the design of MLP networks. One needs a sufficient number of connections in order to be able to learn an important training base. However too many connections lead to a system with many degrees of freedom, and thus the backpropagation algorithm gives poor results. The shared-weight connections is a method to reduce the number of free parameters, while conserving many weights.
The performances are summed up in table1. The best simulation gives 98,2% recognition without reject, with 1,2% fluctuation in relation to the weight's initialization. A good classification rate is obtained even with a large reject threshold, and the rate of ambiguity is very low (< 0,4%).
The most confused characters are (X,Y), (1,I), (O,D) and (4,H) or (4,e).

3. Modular network

3.1. A natural separation

In a first trial, we have separated **arbitrarily** our data base in three sub-bases: digits, upper and lower-cases letters. The structure of the corresponding networks is identical to the global one (see fig.1). We only changed the number of the outputs for each of them.

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 111-116

Each network is trained on its associated training set and it never sees the other characters; it is the same thing for the test base.
Separately, these experts give good performances : 99% for the digit net, 95% for the upper-case one and 99% for the lower-case one.
We want to create a simple cooperation scheme between the three experts; so we use a non-linear perceptron (module n°2), with the outputs of experts as inputs and 49 cells in the output. In the learning phase, the weights of the experts are fixed and we need the whole training set (we present all the characters on the three experts inputs). We hope that the experts will not give a strong answer on a character it never sees.
In fact, most of the confusion arises within (O,C), (8,B), (O,n), (1,I), (X,Y), that is to say either between 2 differents nets, or often in upper-case net. It seems that the cutting up of the base is not optimal as similar characters are not learned in the same expert.

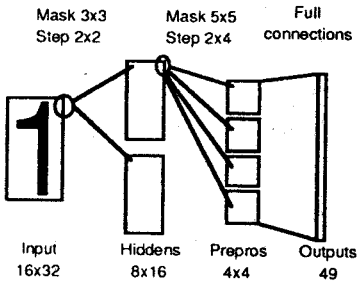

Figure1

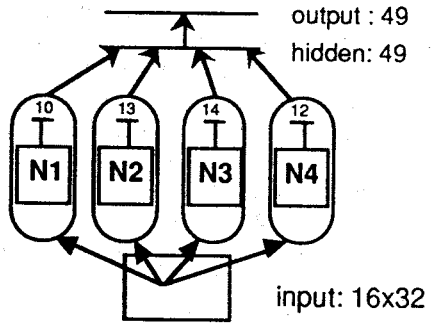

Figure 2

Figure 1: The 8x16 units in the first hidden layer are connected to the input image through a 3x3 mask, and its shifting is a two-dimensionnal 2x2 step. The second hidden layer (named "prepros") has the same connections with a 5x5 mask and a 2x4 step. The connections between prepros and the 49 outputs are full.
Figure 2: Module 4

3.2. Separation in four experts

We tried to separate the data base in 4 sub-bases in respect to the remark above : "big circle" (0,O,Q,8,G,B,C,2,R,P), "little circle" (9,6,3,4,A,e,5,a,b,d,g,h,q), "lines" (I,J,t,1,L,7,E,T,H,f,i,j,r) and "cross" (X,Y,M,W,N,K,S,Z,U,V,m,n). We have grouped the characters confused by the global network, in the same subnet like (X,Y) in "cross" or (I,1) in "lines". Each subnet has been completed by characters (by example 8 and B in "big circle" or 3 and 5 in "little circle") which have been well classified by the global network, in order to obtain roughly the same number of classes.
Each network has a structure similar to that in figure 1, with its corresponding outputs. We trained them as in 3.1. In this case too, we've had good performances for each expert.
We have introduced two cooperation schemes. The first does not use a neural net and is simply the class-max (module n°3). The second is a MLP (module n°4) which has 49 inputs (the values of the expert outputs), 49 output cells and 49 hidden cells. The weights of the experts are fixed and we present the whole test set.

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 111-116

These first results show that the performances are strongly dependent on the cutting up of the data-base, and a "natural" separation is not the best (in our problem). Moreover, MLP nets, for cooperation, cannot transform confused characters into recognized ones. It seems that the choice of an MLP net for cooperation instead of a simple max rule does not really improve the rate of the performances. But it provides a safer separation between the classes, as it allows better saturation. We tried different structures of MLP, without the hidden layer or with different numbers of cells within it and we chose an optimal structure.

4. One expert by class

4.1. First structure

In a first time, we conserve a complex architecture for each expert in order to get a good classification rate (cf Figure 1). We have 2 output cells, (class, no-class). One could be sufficient but we hope the information given by the 2 outputs will be different and will thus improve the cooperation.
For each expert, we create a pseudo-training set : we present sequentially a no-class character with an uncertain choice, and a class character, until every non-class character has been presented. We do not conserve the statistics of the data-base elements, but the advantages are : no saturation of the no-class, extraction of characteristics of the class, and knowing of the global problem.
In that case, the result of each network is larger than 99,8% recognition with the class-max decision. The networks are certainly more complicated than necessary.
We made a cooperation without network as in 3.1 (module n°5). We observed that the outputs (class, no-class) were symetric, so we only conserve the output class. The performances are excellent, but there is a degradation with a high reject threshold. The most confused characters are (X, Y) and (1, I). It seems that the network has real difficulties to discriminate between them.

We tried several neural cooperations (perceptron, MLP), but we cannot obtain more than 92% recognition without reject (module n°6). At the moment, we cannot give any explanations of this low rate.

4.2 More simpler networks

The results obtained in 4.1 are very interesting as this splitting of the problem leads to good performances. However, in an attempt to reduce the complexity of the classification task, this approach is not believable. Indeed, the structure of each expert network is as big as the general one and their learning is very long. So we decided to consider only a perceptron with a 16x32 image as input, 2 output's cells, and full connections.
Learning is a bit different. We saw that there is a strong relationship between the number of weight and the data-base dimension. Só, we artificially reduce our training set : we keep all the characters of the class and we choose uncertainly the others in the rest of the 1500 characters set. The learning algorithm is the same as in 4.1.

Each expert gives bad results, but the value of the two outputs are no more opposite, so it is judicious to use both of them.

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 111-116

The cooperation with class-max (module n°7) gives 77% recognition without reject, so we need another cooperation module to improve the global performances.
We observe the behavior of a cooperation with the perceptron. We compare a network with class as input (module n°8) with an other one with class and no-class as input (98 cells, module n°9). As the performances of the second net are better than those of the first, it is clear that the no-class information is really discriminant. Moreover, during the learning phase, we force the net to learn the characters confused by the experts by presenting them more often than the others.
However, the results of the module n°9 are not good enough in respect to those of the modules n°1 and n°5. So we designed a cooperation network (module n°10, see figure 3) with more informations on the input-layer. Indeed, it contains both the outputs of the experts (98 class and no-class cells), and the 49 cells coding the desired class. The learning phase is the same as for modules n°8 and n°9. During the training phase, as we do not know the desired class, we cannot use this net as it stands. For each character to recognize, we presented successively the 49 possible classes. At each presentation we keep the maximum value of the outputs, and finally, we attribute to this character the class associated to the largest of these maxima. One simulation gave a little more than 97% recognition without reject, but we noticed that the values of the outputs did not saturate. Unfortunately, the computing time is very large,so we are now looking for a way to reduce it.

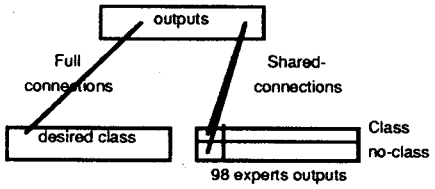

Figure 3 : Module 10. Each boxes contains 49 cells

5. Conclusion

This work shows that, on a multifont character recognition task, the performances obtained with modular networks are the same as those obtained with a single network. In fact this is only true if the expert are good enough and if the splitting of the problem is "well" done. As it is not easy to choose the different modules, it could seem more attractive to build a network per class as moreover, in this case, incremental learning is straightforward. However, this approach is interesting only if each network is smaller than the global one, otherwise the calculation time is not reduced.
We thus tried simple perceptron nets to recognize each class instead of more complicated ones. Of course, the individual performances are less good and a simple cooperation scheme is not efficient to solve the global task. We thus designed a MLP cooperation net which learns with two kinds of inputs : the output of the expert nets and the desired class. This more clever cooperation network significantly improves the preceeding performances but its implementation remains heavy. In conclusion, we can say that this work which has to be pursued is very encouraging in the ability of modular architecture to solve complex tasks. It also stress the need of efficient cooperation scheme

ESANN'1993 proceedings - European Symposium on Artificial Neural Networks
Brussels (Belgium), 7-8-9 April 1993, D-Facto public., ISBN 2-9600049-0-6, pp. 111-116

NETWORKS	Sort of cooper.	PERFORMANCES (%)			
		Rejected	Recognized	Confused	Ambiguous
Module n°1	class-max	0 3,73	98,0 95,5	2,0 0,72	0 0,025
Module n°2	MLP	0 24,27	91 72,74	9 2,98	No cal-culated
Module n°3	class-max	0 3,47	97,4 95,29	2,52 1	0 0,2
Module n°4	MLP	0 2,16	97,43 96,45	2,57 1,23	0 0,15
Module n°5	class-max	0 3,08	97,8 96,19	2,13 0,56	0 0,15
Module n°6	MLP	0 16,38	92,38 82,97	0,48 0,36	0 0,28
Module n°7	class-max	0 17,9	77,82 74,27	22,17 2,59	0 5,19
Module n°8	MLP	0 10,6	86,13 81,43	13,86 2,49	0 5,41
Module n°9	MLP	0 8,59	89,0 84,0	10,9 2,72	0 4,62

Table 1. Comparative performances of the different networks. The first line contains the results without threshold and the second contains the results with a reject threshold (0,7) and a ambiguity threshold (0,1).

Acknowledgement

The simulations of the MLP networks have been done with the simulator SN2 of Neuristics.
The authors (B. Dorizzi and P. Sebire) whish to thank O. Volt for his computer assistance.

References

[BOL91] M. de Bollivier, Application des réseaux connexionnistes à la reconnaissance de formes. Introduction du concept de modularité. Thèse de doctorat, Université Paris XI, 1991

[IDA92] Y. Idan, J.M. Auger, N. Darbel, M. Sales, R. Chevallier, B. Dorizzi, G. Cazuguel, Comparative study of neural networks and non parametric statistical methods for off-line handwritten character recognition, Proceedings ICANN 92, Brighton, September 1992.

[JAC91] R. A. Jacobs, M.I. Jordan, S. J. Nowlan, G. E. Hinton, Adaptive Mixtures of Local Experts, Neural Computation 3, 79-87, 1991.

[LEC90] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R.E.Howard, W. Hubbard , L.D. Jackel, Handwritten digit recognition with a backpropagation network. In D.Touretzky (ed.) Neural Information Processing Systems, 2, Morgan Kaufmann.

[WAI89] A. Waibel, Modular Construction of Time-Delay Neural Networks for Speech Recognition, Neural Computation 1, 39-46, 1989.