ESANN'1996 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-25-26 April 1996, D-Facto public., ISBN 2-9600049-6-5, pp. 1-6

Synaptic Efficiency Modulations for Context Integration The Meta ODWE Architecture

N. PICAN

CRIN-CNRS / INRIA Lorraine
BP 239
F-54506 VANDŒUVRE-les-NANCY Cedex
FRANCE

Abstract : The new neural network architecture named ODWE (Orthogonal Delta Weight Estimator) implements a context dependent behavior by dynamically estimating the synaptic weight variations of a main MLP with regard to context parameters. The main MLP is first trained to modelize the general behavior of the task and then these connections are modulated to adapt to contextual parameters. We present in this paper a new extension of this principle by the generalization of the use of contextual modulation and show its performances in a combinatory problem: the multiplexor problem. We compare these results with other works.

1. Introduction

It is commonly known that Artificial Neural Networks (ANN) have great ability to solve many kinds of tasks, such function approximation, classification, optimization, neurocontrol. Many ANN architectures have been developped to solve such or such task [2]. One of the best known is the multilayered Perceptron (MLP). The MLP architecture has a great ability for generalization in a noisy environment or in non-fulcovered spaces of training examples. The input space of a MLP is often defined by one dimension for each known parameter of the task. In many cases the input space size becomes very large and the internal connectivity increases accordingly.

Our studies [6] showed that it is ofttimes possible to analyze the input space of a problem as a perception space plus a context space. The first architecture developped on this analysis has been the OWE, in which each connection of a MLP (main MLP) fed with the perceptual part of an input pattern, is estimated by another MLP named OWE ("Orthogonal Weight Estimator") fed by the contextual part of an input pattern. We extended the OWE architecture to the ODWE architecture ("Orthogonal Delta Weight Estimator") in which one ODWE computes a variation of each connection in the main MLP. The problem becomes, in the first phase, to train the main MLP to solve the task all contexts being joined (general task). In a second phase, a set of MLPs (ODWEs) fed with context parameters is trained to estimate the variations of each connection value of the main MLP wrt[1] the inputs in the context space to adapt this general task [5].

1. with respect to

ESANN'1996 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-25-26 April 1996, D-Facto public., ISBN 2-9600049-6-5, pp. 1-6

Two problems appeared in the use of the ODWE architecture. The first one is that the use of one ODWE for each connection of the main MLP is sometimes superfluous. The second one is the opposite *i.e* sometimes, one level of abstraction, given by an ODWE, is unsufficient.

The content of this paper proposes a solution of these problems and presents some results on a combinatory problem.

2. Superfluous ODWE

The first step of the training process to the ODWE architecture is to train the main MLP that will compute, if taken alone, a general solution to the task or, in other words, the general behavior. During this phase each connection originating from the input space (respectively from neurons in hidden layers) of the main MLP, captures the mean feature on the context of the perception space (respectively of the internal representation of the input space) wrt the task. At the end of this phase, the mean feature will be the exact feature if this feature does not depend on the context variations.

The end of the training phase is commonly chosen with regard to desired mean error over a test corpus. Of course it is not yet possible with our architectures because the context is not taken into account, so the mean error is always greater than the desired one. Another way to detect the end of training is to see that the mean of the error gradient, taken on one epoch, for all connections is null. This is the test of stationnarity of gradients. If we apply this test for one connection it can be said that the connection value does not change anymore and has captured the mean feature. Now, in order to know if this connection needs an ODWE neural network to compute the variations of the mean feature wrt context variations, it is necessary to know if the connection value reflects a mean feature or an exact feature which does not depend on context variations. This knowledge is given by the variance of the gradient. In classical MLPs the gradient variance gives only the importance of the input/output noise on the connection. Here, because some inputs of the main MLP have been removed, the gradient variance of a connection also reflects the context dependency of the connection value.

This analysis clearly defines a general algorithm for Incremental ODWE architecture.

In this algorithm, W_{ij} is the connection value, x is an input of the main MLP taken in the X perception space, φ is a context input, μ_{φ} is the conditional probability distribution over the perception space X given the context φ, $\delta W_{ij}(\varphi)$ is the output of the ODWE attached to ij connection fed by φ, $\Lambda(x)$ is the error function, $\nabla_{W_{ij}}$ denotes the gradient wrt W_{ij}, $\beta \in [0, 1]$ is the rate of ODWE attachment, $\varepsilon \ll 1$, λ is the learning rate, E denotes the expectation and V the variance.

ESANN'1996 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-25-26 April 1996, D-Facto public., ISBN 2-9600049-6-5, pp. 1-6

(a) propagate all inputs $x \in X$

$W_{ij} \equiv W_{ij} + \delta W_{ij}(\varphi)$ for connections attached with an ODWE

(b) for each connection of the main MLP

compute $E_{\nabla_{ij}} = \int_X \nabla_{W_{ij}} \Lambda(x)\, \mu(dx)$

(c) update connections

$W_{ij} \leftarrow W_{ij} - \lambda E_{\nabla_{ij}}$

(d) if $\sum_{i,j} E^2_{\nabla_{ij}} > \varepsilon$ goto (a)

(e) for each connection ij of the main MLP

compute $V_{\nabla_{ij}} = \int_X \nabla_{W_{ij}} \Lambda(x)^2 \mu(dx)$ ($E^2_{\nabla_{ij}}$ is neglected)

(f) for each ij: $V_{\nabla_{ij}} > (1-\beta) \cdot max_j(V_{\nabla_{ij}})$ attach a ODWE to W_{ij} connection

(g) train each ODWE(ij) by using for each φ the desired output given by :

$E_{\nabla_{ij}}(\varphi) = \int_{X(\varphi)} \nabla_{W_{ij}} \Lambda(x)\, \mu(dx, \varphi)$

(h) propagate all inputs $(x, \varphi) \in X \times \phi$ by using

$W_{ij} \equiv W_{ij} + \delta W_{ij}(\varphi)$ for connections attached with an ODWE

(i) if test on mean error is not reached increase β and go to (f)

FIGURE 1. General algorithm for Incremental ODWE.

This algorithm presented above is based on the batch backpropagation learning algorithm with a constant learning rate. We have changed it into an optimized stochastic gradient descent algorithm by using the moving means to compute the expectations and variances and the On-line learning algorithm [4] for ODWE training.

3. The Meta ODWE architecture

In the opposite direction of the problem of superfluous ODWE we mentioned the problem where the abstraction given by the level of ODWE in unsufficient. This problem is due to functions $\varphi \rightarrow W_{ij}(\varphi)$ learned by ODWE, which are very complex. The same kind of problem can be observed as one tries to solve the task with a classical MLP. This problem is tackled using the same principle, developped for the main MLP, for the ODWEs themselves. Then the architecture becomes the Meta ODWE where the ODWEs at a high level compute the variations of the connection values of ODWEs in a lower level that compute... the variations of the connection values of the main MLP.

This principle can also be adapted to create a Meta OWE architecture in which OWEs in high level compute the connection values of OWEs in lower level that compute... the connection values of the main MLP.

ESANN'1996 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-25-26 April 1996, D-Facto public., ISBN 2-9600049-6-5, pp. 1-6

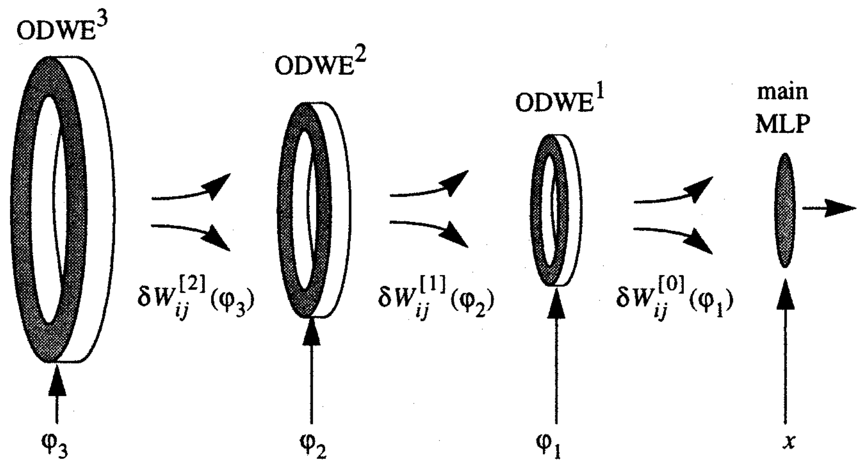

FIGURE 2. Example of a Meta ODWE Architecture of level 3 where the torus represent the sets of ODWEs.

4. Discussion and results

The solutions of the two problems depicted in previous paragraphs have be mixed in an Incremental Meta ODWE architecture. This shows the interest of an ODWE approach, since the solution is impossible for the Meta OWE architecture.
In the following, the terms OWE and ODWE will respectively reflect Meta OWE and Incremental Meta ODWE.
We tested ours architectures on a non trivial problem of multiplexor. In this problem, the task is to determine the value of a bit in binary data given its address.

Example: *8 bits multiplexor problem*

	7	6	5	4	3	2	1	0
input binary data :	1	0	1	**1**	0	1	0	1
input address :	1	0	0	= 4				
desired output :	1							

It is a combinatory problem in which the number of combinations is $2^{n + \log_2(n)}$, where n is the number of bits in binary data.
The analysis of the problem input space clearly defines perception space as the byte data and context space as the address. To emphasize the properties of our algorithm we choose a simple Perceptron architecture with a bias for all MLP in the Meta structure and an identical context spaces with $\varphi_i = \varphi = \text{address}$.
In the following table we present the results of training OWE and ODWE[1] architecture on the 4, 8, 16 bits multiplexor problem and compare them with the results obtained with a MLP with one hidden layer (fed with data and address) and two algorithms of

1. Table notation "*a/b/c*": *a* in level 1, *b* in level 2, *c* in level 3

progressive construction rule: GA (Genetic Algorithm) [8] and a growing ANN algorithm (gANN) developped in [1] that dynamically creates a pair of cells representing condition/action rule for a classe of input patterns.

TABLEAU 1: 4 bits multiplexor (64 combinations)

Algorithm	Number of patterns presented to obtain 100%	Number of "rules"
GA	10,000 (99.4%)	400 rules
gANN	700	12 rules
MLP	3,900	10 neurons in the hidden layer
OWE	31	5 / 20 / 0 OWEs
ODWE	794	3 / 1 / 0 ODWEs

TABLEAU 2: 8 bits multiplexor (2,048 combinations)

Algorithm	Number of patterns presented to obtain 100%	Number of "rules"
GA	30,000 (90%)	400 rules
gANN	6,000	25 rules
MLP	34,000	30 neurons in the hidden layer
OWE	1,800	9 / 36 / 0 OWEs
ODWE	6,700	8 / 25 / 0 ODWEs

TABLEAU 3: 16 bits multiplexor (1,048,576 combinations)

Algorithm	Number of patterns presented to obtain 100%	Number of "rules"
GA	120,000 (90%)	1600 rules
gANN	85,000	49 rules
MLP	3,000,000 (99.8%)	100 neurons in the hidden layer
OWE	27,400	17 / 85 / 425 OWEs
ODWE	176,000	17 / 55 / 274 ODWEs

Three important conclusions can be drawn from these results. The first one is the power of OWE and ODWE to solve this difficult problem. The second one is the comparison between OWE and ODWE that shows that OWE learns with less example presentations than ODWE. But, regarding computation time, ODWE is the fastest because the incremental architecture provids only few ODWE attachment against a necessary full attachment of all OWEs in the OWE architecture. We do not say that the ODWE architecture gives an optimal architecture but it is optimal with regard to OWE. The third conclusion is very important with regard to a real time computation with ODWE and can be depicted in two points:

- All ODWEs in a given level can compute in parallel (as for OWE).
- The main MLP in a ODWE architecture can compute very fast a general solution to the task, if it is necessary, and adapt this solution wrt context in the next cycles.

ESANN'1996 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-25-26 April 1996, D-Facto public., ISBN 2-9600049-6-5, pp. 1-6

The OWE and ODWE have been implemented on Intel Paragon computer with 60 i860 nodes. Then the computation time in each ODWE level is approximately divided by the number of nodes.

5. Conclusion

These results first show the good performances of our models and more generally, the validity and the interest of the orthogonal approach that we have been studying for three years [5], with equal success, on industrial problems [7].
Beyond this aspect, we also want to underline the underlying meaning of this kind of architecture. For a long time, neurobiological studies have shown the existence of neurons performing filtering on data with classical feedforward links. This kind of connectivity is at the root of most classical ANN models. More interestingly, other neurobiological studies stress the existence of modulator neurons whose connectivity is orthogonal to the main information flow mentioned above. These neurons whose role increases in phylogenesis are at the bases of multisensory integration and context-dependent behavior [3]. We think that our approach can orient artificial connectionist models towards a better modeling of this kind of abilities.

References

[1] **Blanchet P.**: "Une architecture connexionniste pour l'apprentissage par l'experience et la representation des connaissances", PhD thesis, Paris XI University, LIMSI, Orsay. December 8, 1992.

[2] **Hertz J., Krogh A. and Palmer R.**: "Introduction to the Theory Of Neural Computation". Addison Wesley Ed., Lecture Notes of Santa Fe Institute, Vol I. March 1992

[3] **Miller R.** : "Cortico-Hippocampal Interplay and the representation of Contexts in the Brain". Springer Verlag Berlin Heidelberg, 1991.

[4] **Pican N., Fort J.C., Alexandre F.**: "An On-line Learning Algorithm for the Orthogonal Weight Estimation of MLP". Neural Processing Letters, Vol 1, No 1,21-24, 1994.

[5] **Pican N.**: "Approche Statique et Dynamique de la modulation des Efficacites Synaptiques dans les Réseaux de Neurones". PhD thesis. Henri Poincare University, Nancy. January 18, 1995.

[6] **Pican N., Alexandre F.** : "How OWE Architectures encode Contextual Effects in ANNs". To appear in Mathematics and Computers in Simulation. 41 (5-6) July 1996.

[7] **Pican N., Alexandre F., Haton J.P., Bresson P.** : "NeuroControl of Temperatures with OWE Neural Network Architecture in Continuous Steel Annealing Furnace". Industrial session of ICANN'95. October 9-13, 1995.

[8] **Wilson W. Stewart** : "Classifier Systems and the Animat Problem", in Machine Learning, Vol. 2, 199-228, 1987.