Towards active and progressive learning in Artificial Neural Networks

Nicolas SZILAS
LIFIA-ACROE-IMAG
INPG - 46, av. Félix Viallet - 38031 Grenoble Cedex - France
Nicolas.Szilas@imag.fr

Introduction

This paper does not present any algorithm nor theoretical results concerning active learning. Rather, we propose a broad overview of the use of a learning strategy in the field of supervised neural networks. By learning strategy we mean a technique that handles the order of presentation of the patterns to be learned.

We distinguish two types of works: informative learning and progressive learning, which are presented respectively in the first and second section. In the third section, we introduce the concept of active and progressive learning as a challenge for further research.

New algorithms based on this principle, that are still to be designed, should significantly improve the learning abilities of present learning algorithms.

Review of Informative Learning

Most active algorithms applied to neural networks learning can be grouped together, since they are all based on the same principle: the maximization of the information coming from the environment.

Several motivations exist for selecting the most informative examples from the environment:

- Firstly there are some redundancy in most training sets that imply a lost of time while reaching a good solution. Thus, by choosing the most significant patterns on can improve the speed of convergence (Allred & Kelly 1989)(Strand & Jones 1992)(Plutowski & White 1993)(Atlas *et al.* 1990)(Cachin 1994). The corollary of this is that it allows to reduce the final error (Hwang, Choi, & Marks 1991).
- Secondly, the active selection of examples can improve the generalization performance of the obtained network (Biehl & Schwarze 1992)(Röbel 1994)(Zhang 1994)(RayChaudhuri & Hamey 1995)(Krogh & Vedelsby 1995). Effectively, it avoids some too specific solutions to be obtained.
- Thirdly, in some applications, the acquisition of an example is costly, because it involves a complex measurement process (Hwang, Choi, & Marks 1991)(Atlas *et al.* 1990). In such a case, active learning is interesting too minimize the number of samplings of the environment.

Depending on the algorithms, the learning sessions are organized in different ways:

- Given a training set, some algorithms incrementally construct an optimized subset (Plutowski & White 1993)(Röbel 1994)(RayChaudhuri & Hamey 1995)(Krogh & Vedelsby 1995).
- Other algorithms weight the influence of each example of the training set, either by skipping some examples (Haffner 1988)(Allred & Kelly 1989)(Strand & Jones 1992), or by modifying the frequency of presentation of patterns (Munro 1992)(Cachin 1994).
- In the case of *query learning*, each learned example is successively obtained by sending an input to the environment which yields the corresponding output.

At last, several criteria to select the most informative patterns of the environment, empirically or theoretically based, have been proposed:

- The most used criterion is the network's *error*: the examples for which the network gives the highest error are preferentially learned (Haffner 1988)(Allred & Kelly 1989)(Strand & Jones 1992)(Munro 1992)(Cachin 1994)(Röbel 1994)(Zhang 1994).
- A much more sophisticated criterion has been developed in (Plutowski & White 1993), that consists in choosing the example that maximizes the decreasing of the error calculated over the whole database.
- For classification problems, it is interesting to choose the examples that are not well classified, that is the examples near the borders of classes (Hwang, Choi, & Marks 1991)(Biehl & Schwarze 1992). This criterion is related to the "error criterion", since a pattern near a border necessarily gives a high error. It presents the advantage to be applicable for query learning, when the desired output is not given for selecting an example.
- However, the "border criterion" only gives a subset of examples that should be selected, because back-propagation algorithm tends to be "overconfident" in unexplored regions (Atlas *et al.* 1990). Thus, a more sophisticated criterion consists in selecting examples in a *uncertainty* region, which can be estimated as following: two networks are trained on already selected examples and on "background" examples randomly chosen in the input space; the first network

associates to these background inputs the first class, and the second the second class. The uncertain inputs are those for which the two networks give a different class.
- Another quite interesting approach to select examples consists in using several networks trained on many subsets of the training set (RayChaudhuri & Hamey 1995)(Krogh & Vedelsby 1995). By evaluating for all the examples the disagreement between these networks, the most ambiguous examples can be selected, considered as the most informative patterns. Note a similarity between this "committee" approach and the uncertainty based approach described above.

From a cognitive point of view, this idea, that the system preferentially learns the most informative parts of the environment is quite consistent with psychological data: indeed, the possibility of *asking a question* about an unknown fact is essential in human learning; on another level, it has been shown that tactile exploration especially focuses on highly informative regions (Hatwell, 1986), as in visual exploration. Consequently, human and machine learning share the same property: they are based on a principle of *economy* of information processing; *pure* random or exhaustive exploration is excluded.

Review of progressive learning

In the field of connectionism, there is another group of studies that concerns the effect of the strategy of patterns presentation.

It consists in considering that the learning system is not learning an environment as a whole and at once, but is immersed in a temporal context where the final task is to learn the environment, but intermediate tasks may be created to achieve learning. This temporality must be considered as an advantage, since it gives the possibility to make a *progressive* learning.

There are very few studies on that aspect of learning in neural networks, although it was evoked some years ago ((Le Cun, 1987), pp. 175-176).

Two early experimental studies from 1988 studied this aspect for back-propagation networks. In (Wieland & Leighton 1988), the idea of *shaping* coming from behaviorist psychology is applied to distinguish images of "A", "B", "C" and "D" independently on rotation. Progressively adding more and more difficult new examples into the training set improved the speed of convergence by a factor of *4*. Unfortunately, the precise schedule of pattern presentation is not given (only an abstract has been published).

In (Jacobs 1988), the first problem consists in learning the associations 0.49-> 0.1 and 0.51-> 0.9. The classical learning of this task requires 18371.7 epochs in average. In the progressive learning condition, the first task is to learn the associations 0.4-> 0.1 and 0.6-> 0.9, the second task the associations 0.45-> 0.1 and 0.55-> 0.9, the third task the associations 0.48-> 0.1 and 0.52-> 0.9, the fourth task the associations 0.49-> 0.1 and 0.51-> 0.9 and at last the fifth task is the original task. With such a scheme, the *total* number of epochs is only 4447.1 in average. The second problem shows similar results with the learning of sinusoidal oscillations.

An interesting experiment is also described in (Fahlman 1991) where the recurrent cascade correlation architecture is used to learn the Morse code. Learning is achieved by presenting the patterns from the easiest ones (the shortest sequences) to the most difficult ones (the longest sequences).

The study of (Elman 1993) is the most famous study concerning progressive learning. The problem is a grammatical prediction task: sentences generated by an artificial grammar are presented to a Simple Recurrent Network which must predict at each time the probability of occurrence of each word in the vocabulary. When the corpus of sentences is presented as a whole to the network, learning fails. Conversely, when the training set is divided into two subsets, one containing the easy sentences (short ones) and the other the complex sentences (long ones), and when the proportion of complex sentences in the current training set is progressively increased, then the learning succeeds.

More recently, many experiments have been carried out by Ian Cloete and Jacques Ludik on learning strategies. They tested:
- *Combined Subset Training*, introduced in (Cotrell & Tsung 1991), which consists in successively doubling the size of the training set of randomly chosen patterns,
- *Increased Subset Training* (Cloete & Ludik 1994) where instead of multiplying the size of the training set by two, it is incremented by a fixed number,
- *Increased Complexity Training* (Cloete & Ludik 1993) which is like *Combined Subset Training*, except that the patterns are selected in a specific order of increasing complexity, evaluated by the length of the sequences in that case,
- *Incremental Increased Complexity Training* (Ludik & Cloete 1994) which is like *Increased Complexity Training* except that the training set is progressively incremented (four by four, for example),
- *Delta Training Strategies* (Ludik & Cloete 1995) which consists in ordering the examples according to some "discriminability" criterion, calculated from distance measurements in the input space.

In the experiments, the last four strategies improve the speed of convergence compared to fixed set training. *Incremental Increased Complexity Training* gives quite good results. When no complexity measure exists, *delta Training Strategies* also gives good results.

At last, in our experiments (Szilas & Ronco 1995), we show that progressive learning applied to feedforward nets not only improves the speed of convergence but also the rate of success. For the XOR problem, with classical learning, 22 trials failed out

100, whereas with progressive learning (two examples first, and then the four examples), learning only fails for 10 trials. Such progressive learning was tested for a much more complex task: the two spirals classification task. With a classical back-propagation networks, learning succeeds for only 4 out 20 trials. If the patterns are presented in a specific order, by beginning with the external parts of the spirals and then the inner parts, learning succeeds for 20 out 25 trials.

From all these experiments, we conclude that grading the difficulty of the task can radically improve the performance. However, all the experiments are based on a difficulty criterion that is based on a priori knowledge that the user has, except for *Increased Subset Training*, for which difficulty is only roughly controlled by the number of examples, and the *Delta Training Strategies*, whose difficulty criterion is not general and is independent on the network representation. For complex problems, knowledge about how the difficulty can be increased is not necessarily available. Furthermore, for all these algorithms, some quite sensitive parameters must be carefully set up by hand, to decide when the current training set has to be increased and how many patterns must be added.

Thus most current progressive learning algorithms are not active in an autonomous way. The challenge is to find a general enough difficulty criterion to make the algorithms independant on the task and on the user.

Active and progressive learning

Until now, we have considered informative learning and progressive learning as two classes of active (existing or future) learning algorithms. It is however interesting to consider "informative" and "progressive" as two *dimensions*, two features caracterizing all the active algorithms. The informative dimension represents the maximization of information coming from the environment; the progressive dimension represents the progressive raising of the difficulty of the information coming from the environment.

We then realize that numerous algorithms that we have classified as informative have in fact a progressive dimension: when the examples are added one by one in the training set, as in (Atlas *et al.* 1990)(Hwang, Choi, & Marks 1991)(Plutowski & White 1993)(Röbel 1994)(RayChaudhuri & Hamey 1995)(Krogh & Vedelsby 1995), the difficulty of the task does increase with time (Röbel 1994). Thus, the increase of performance of all these algorithms is not only caused by the fact that only most significant patterns are learned, but is also due to an increasing difficulty. *If* those algorithms selected an informative subset and then learned it, they would be purely informative. The algorithms that skip examples in the training set (Haffner 1988)(Allred & Kelly 1989)(Strand & Jones 1992) are purely informative.

Thus, in a certain way, some active and progressive algorithms exist, but they are only weakly progressive, since the implicit criterion of difficulty is the number of non redundant examples, criterion that quickly gives a complex set.

Although the notion of difficulty that we are discussing is still rather intuitive, we propose a graph representing the difficulty of the task according to time, for different kind of algorithms (Fig. 1). In current active learning algorithms, the difficulty is increasing too quickly; the purpose is to find a way to smoothly increase the difficulty.

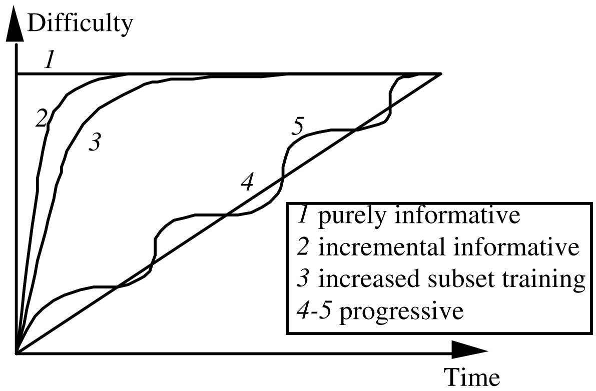

Figure 1

In another graph, we have represented the two dimension space in which the different algorithms can be represented (Fig. 2). In abscissa is represented the progressive dimension and in ordinate the informative dimension. The autonomous algorithms are represented by a dot whereas the algorithms where the human intervenes by giving a priori information are represented by a square. The points *E* and *F* represent algorithms that are still to be designed.

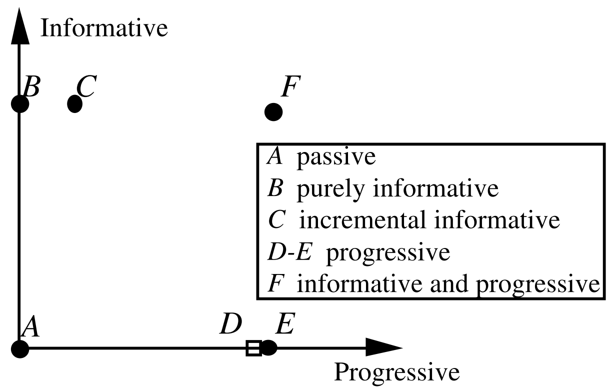

Figure 2

Conclusion

The progressive dimension that is the central point of this paper must not be only regarded as a way to speed up learning. Indeed, as shown in (Elman 1993) and (Szilas & Ronco 1995), progressive learning does make the learning succeed. Thus, the real challenge of active progressive learning is to extend the class of problems that can be currently learned by neural networks.

Two directions must be followed to achieve this goal. First, an effort to formalize progressive learning must be made. Second, some interesting ideas can be found in the human cognitive system, that achieves a kind of progressive learning in many tasks: problem solving, motor learning, etc.

Note that in the field of connectionism, some interesting experiments have been carried out concerning the *transfer* of an already learned skill to a new task (Sharkey & Sharkey 1994)(Thrun & Mitchell 1994). This issue is particularly related to progressive learning.

References

Allred, L.; and Kelly G.E. 1989. Supervised Learning Techniques for Backpropagation Networks. In Proc. IJCNN, vol. I: 721-728.

Atlas, L.; Cohn, D.; Ladner, R.; El-Sharkawi, M.A.; Marks II, R. J.; Aggoune, M.E.; and Park, D.C. 1990.Training Connectionist Networks with Queries and Sampling. In D. S. Touretzky (ed.): *Advances in Neural Information Processing Systems 2*: 566-573, San Mateo: Morgan-Kaufmann.

Biehl, M.; and Schwarze, H. 1992. On-Line Learning of a Time-Dependant Rule. *Europhysics Letters*, 20(8): 733-738.

Cachin, C. 1994. Pedagogical Pattern Selection Strategies. *Neural Networks*, 7(1): 175-181.

Cloete I.; and Ludik, J. 1993. Increased Complexity Training. In Proc. IWANN, Sitges, Spain.

Cloete I.; and Ludik, J. 1994. Incremental Training Strategies. In Proc. ICANN, Sorrento, Italy.

Cotrell, G.W.; and Tsung, F.S. 1991. Learning Simple Arithmetic Procedures. In J. A. Barnden, J. B. Pollack (eds): *High-Level Connectionist Models*, in the series Advances in Connectionist and Neural Computation Theory, 1: 305-321.

Elman, J.L. 1993. Learning and development: the importance of starting small. *Cognition*, 48: 71-99.

Haffner, P.; Waibel, A.; Sawai H.; and Shikano K. 1988. Fast Back-Propagation Learning Methods for Neural Networks in Speech. ATR Tech. Report TR-I-0058.

Hatwell, Y. 1986. *Toucher l'Espace*. Presse Universitaire de Lille.

Hwang, N.J.; Choi, J.J.; Oh, S.; and Marks II, R. J. 1991. Query-Based learning applied to Partially Trained Multilayer Perceptrons. *IEEE Trans. on Neural Networks*, 2(1): 131-136.

Jacobs, R.A. 1988. Initial Experiments On Constructing Domains of Expertise and Hierarchies In Connectionist Systems. In Proc. Connectionist Models Summer School: 144-153, 1988.

Krogh, A.; and Vedelsby, J. 1995. Neural Networks Ensembles, Cross Validation, and Active Learning. In Proc. NIPS-7.

Le Cun, Y. 1987. Modèles connexionistes de l'apprentissage. PhD diss., Université Paris VI.

Ludik, J.; and Cloete, I. 1995. Training Strategies for Architecture-Specific Recurrent Neural Networks. *S.A. Comuter Journal*. Forthcoming.

Munro, P.W. Repeat Until Bored: A pattern Selection Strategy. In Proc. of NIPS-4: 1001-1008.

Plutowski, M.; and White, H. 1993. Selecting Concise Training Sets from Clean Data. IEEE Trans. on Neural Networks, 4(2).

RayChaudhuri, T.; and Hamey, L.G.C. 1995. An Algorithm for Active Data Collection for Learning - Feasibility Study with Neural Networks. Tech. Report, 95-173C. Dept of Computing, School of MPCE, Macquarie Univ., Australia.

Röbel, A. 1994. Dynamic Pattern Selection for Faster learning and Controlled Generalization of Neural Networks. In Proc. ESANN'94, Brussels: 187-192.

Sharkey, N.E.; and Sharkey, A.J.C. 1993. Adaptive Generalization. *Artificial Intelligence Review*, 7: 313-328.

Strand, E.M; and Jones, W.T. 1992. An active Pattern Set Strategy for Enhancing Generalization While Improving Backpropagation Training Efficiency. In Proc. IJCNN, Baltimore, vol. I: 830-834.

Szilas, N.; and Ronco, E. Action for learning in non-symbolic systems. In Proc. European Conference on Cognitive Science, Saint-Malo, France: 55-63.

Thrun, S.; and Mitchell, T.M. 1994. Learning One More Thing. Tech. Report, CMU-CS-94-184, School of Computer Science, Carnegie Mellon Univ.

Zhang, B.T. 1994. Accelerated learning by active example selection. *International Journal of Neural Systems*, 5(1): 67-75.