Speed Learning by Adaptive Skipping: Improving the Learning Rate of Artificial Neural Network through Adaptive Stochastic Sample Presentation

R. Manjula Devi
Assistant Professor
Kongu Engineering College
Perundurai

S.Kuppuswami
Principal
Kongu Engineering College
Perundurai

ABSTRACT

The basic idea of this paper is to increase the learning rate of a artificial neural network without affecting the accuracy of the system. The new algorithms for dynamically reducing the number of input samples presented to the ANN (Artificial Neural Network) are given thus increasing the rate of learning. This method is called as Adaptive skipping. This can be used along with any supervised Learning Algorithms. The training phase is the most crucial and time consuming part of an ANN. The rate at which the ANN learns is the most considerable part. Among the factors affecting learning rate, the Size of the training set (no. of input samples used to train an ANN for a specific application) are considered and how the size of the training set affects the learning rate and accuracy of an ANN are discussed. The related works done in this field to reduce the training set are reviewed. The new Adaptive Skipping which dynamically says how many epoch the input sample has to skip depending upon consecutive successful learning of that input sample are introduced.

The algorithm and the steps to train an ANN using the new approach are given and also how the speedup of learning are tested and briefly discussed. The test results are also analyzed. Finally the future works and ideas in this area are discussed. The experiment is demonstrated with the help of a simple ANN using Adaptive skipping along standard Backpropogation for learning.

General Terms

Artificial Neural Network, Supervised Learning Algorithm.

Keywords

Adaptive Skipping, Learning rate, Accuracy, Training set, Backpropogation Algorithm.

1. INTRODUCTION

Neural networks, as name indicates, is a network of neurons i.e. human Brain. The human brain is the most intelligent system in the world, since it has the ability to decide on the given input to give the appropriate output. Thus the simulation of the human brain will lead to develop an artificial intelligent system. Thus the concept of artificial neural networks evolved.

Artificial neural networks have received substantial attention as robust learning models for tasks including classification. To make an artificial neural network to perform specified task first the system has to be trained just as a child is taught. Once a system learns to do the task it will be able to solve any given problem related to the task, even though it hasn't encountered such a problem before.

But the learning rate of a neural network is the problematic section. The factors affecting the learning rate of ANN are,

- Size of the training set
- Efficiency of the learning algorithm
- Network topology
- Type of problem in hand

In particular, the measure to which the training set represents the underlying distribution influences ultimate classification accuracy. Over fitting the training data is often detrimental to generalization. In theory, amassing an infinite training set would provide an exact measure of test accuracy (complete representation of the data distribution) and discourage over fitting. Hence, it is desirable to incorporate as large a training set as possible into the learning phase. However, training on very large data sets is problematic, as training time tends to increase more than linearly with the size of the training set. The time required to converge on large data sets can be prohibitive. Thus decreasing the size of the training set can reduce the training time, but care should be taken to see the accuracy is not affected a lot.

2. RELATED WORK

Numbers of works were done to reduce the training set to achieve faster learning and maintaining generalization. Hush introduced a method for improving the learning rate of the backpropagation algorithm using gradient reuse algorithm. Zhang created a training set by selecting only critical examples and then expand this set if necessary for proper convergence.

Along with these techniques, Owens trains a committee of networks, each network learning from a distinct (balanced) subset of the training data. However, while this can improve training time and generalization, it results in a much more complex solution involving several networks instead of one. This technique's training time is reduced at the expense of testing time. In problem domains where a large amount of high-dimensional data is being classified, such solutions introduce a new problem by slowing down classification.

Notably, the SET method presented by Michael E. Rimer, Timothy L. Andersen and Tony R. Martinez of Brigham Young University achieves a training speedup of up to 42.78% on the data tested with no detectable loss in generalization. They also gave another method called N-Skip method also achieves the speedup with the cost of bit loss of generalization and accuracy when compared to standard method. This method, even though skips the input sample n epochs depending on the error in output due to the input sample, it constantly and randomly says how many times to skip. So this method is not fully adaptive

3. NEW APPROACH

Here the basic idea is "when the network classifies a sample correctly, don't present it again for n epochs (Epoch is one complete cycle of presenting all input samples in the training set once).

Dynamically changing the value of n depending upon the error value of the given input i.e. how the system is learning at present will make the system more adaptive to the present learning situations.

Thus Sample presentation is determined exclusively by the ability of the network to learn. This results in a large reduction in training time through selectively "pruning" correctly classified samples from the training set to exclude their (redundant) presentation to the network each epoch. In other words, only the samples currently affecting the learning process are presented.

Here an effective method is dynamically given to change the value of n in n-skip, which finds the probability of presenting the input sample to the system

3.1 Adaptive Skipping

The value of n should not be fixed. First n is initialized to zero for a particular sample xi. Then for each time the input xi becomes correct, increment the value of n by c-(which is the skipping factor-it should be an integer).As the value of c increases the skipping will increase which result in speedup of learning process.

But there is the danger of loss of accuracy if the value of c is too high. So care should be taken in choosing the value of c.

3.1.1 Algorithm for calculating n

```
Algorithm: Adaptive_skip_basic
{
        If (|ti – oi| < dmax)
                //ti ->target output;
                // oi -> actual output of ith input;
                // dmax ->error threshold.
        then
            n = n + c   //c ->skipping factor
}
```

Note: |ti – oi| give the error value of input sample xi

Thus the probability of presenting the sample xi given by

P(xi) = 0; if xi is correct and number of epoch is less than n.

=1; Otherwise.

Skipping may result in loss of accuracy. Thus to avoid more skipping , the skipping are performed only if |ti – oi| < dmax/2 .For (dmax/2) < |ti – oi| < dmax ,we can calculate the P(xi) by SET method.

Thus the inputs whose |ti – oi| approaching dmax will have the probability to be presented rather than skipping totally. This will increase the accuracy of the system.

One more scenario should be considered here, if the |ti – oi| value is greater than previous epoch's |ti – oi|, but still less than dmax/2 .this tells that |ti – oi| may cross dmax/2 soon. So at this stage, if we increase the number of skips, it may lead to keep the sample which is to be presented, away from the system for a long time. So we decrease the n value by g (g -> decrement factor).More the value of g is more the possibility of accuracy.

Thus the Adaptive Skipping Algorithm changes as follows:

```
Algorithm: Adaptive_skipping
{
  If (|ti – oi| < dmax/2 )
  then
        If (previous Epoch's |ti – oi|< |ti – oi|)
        then
            n = n - g   //g -> decrement factor
        else
            n = n + c   //c ->skipping factor
}
```

Thus the probability of presenting the sample xi given by

P (xi)= 0 ; if |ti – oi| < dmax/2 and number of epoch is l less than n.

= |ti – oi|/ dmax; if dmax/2 <|ti – oi| < dmax.

=1; Otherwise.

3.2 Steps to learn by Adaptive Skipping

1. Present an input sample from training set.
2. Calculate prob and n.
3. Repeat through step 1 for all input samples in the training set (1 epoch).
4. Prepare the new training set for next epoch.
5. Repeat through step 1 until all the inputs have their error less than the threshold error.

4. EXPERIMENT AND RESULTS

A simple ANN are created, which is trained by using standard backprob algorithm. This ANN basically can learn linear equations.

The system are trained to learn a linear equation $y=3x-4$ by using the training set as follows.

Input Sample(xi)	Target Output(ti)
-2.00	-10.00
-0.50	-5.50
0.0	-4.00
1.0	-1.00
2.0	2.00

It needed 5 epochs to learn the equation correctly. Thus 25 samples where presented since for each epoch all the samples to be presented. The same net are trained using the new Adaptive Skipping algorithm along with standard back propagation algorithm. The skipping factor c = 2 and DMAX =0.01 are set.

The net were able to trained in 5 epochs but only 22 input samples are neeeded.

4.1 Time calculation

The approximate formula for time calculation has been derived. The variables which we used are,

N_t - Number of samples in training set

E - Number of epochs

n - Number of samples skipped

N - Number of samples presented by standard method ($E * N_t$)

NL - Number of layers in the network

NN_l - Number of nodes in l^{th} layer

4.1.1 Time for standard method

$$\sim \mathbf{NT_i} \qquad \text{-------------------------(1)}$$

where,

$$T_i = 4\, NN_1 + (NL-1)(NN_l)[(NN_l - 1) + 7] + 4NN_{NL} + (NL-2)(NN_l)(NN_l + 1) + 5(NN_l)(NL)(NN_l - 1) + 6$$

4.1.2 Time saved by adaptive skipping:

$$\sim \mathbf{(nT_i - T_{extra})} \qquad \text{-------------------------(2)}$$

4.1.3 Time for Adaptive Skipping

$$\sim \mathbf{NT_i - (nT_i - T_{extra})} \qquad \text{-------------------------(3)}$$

where,

$$T_{extra} = (N-n) + EN_t$$

For our Experiment…

$N_t = 5$

$E = 5$

$n = 3$

$N = 25$

$NL = 2$

$NN_1 = 1$

$NN_2 = 1$

It was calculated from the above formulas,

$T_i = 33$

$T_{extra} = 47$

Thus, **time by standard method** = NT_i

= **825**

Time saved = $nT_i - T_{extra}$

= **52**

Time taken by our method = $NT_i - (nT_i - T_{extra})$

= **773**

4.2 Test results for accuracy

TEST I:

Table 1. Results obtained in Test I

Input	Output	Output obtained by standard method	Output obtained by adaptive skipping
2	2	2.00	2.00
3	5	5.00	5.00
10	26	26.00	26.00
-23	-73	-73.00	-73.00
-9	-31	-31.00	-31.00

ACCURACY OF STANDARD METHOD: 100%

ACCURACY OF ADAPTIVE SKIPPING: 100%

TEST II:

Table 2. Results obtained in Test II

Input	Output	Output obtained by standard method	Output obtained by adaptive skipping
-11	-37	-37.00	-37.00
-3	-13	-13.00	-13.00
12	32	32.00	32.00
5	11	11.00	11.00
7	17	17.00	17.00

ACCURACY OF STANDARD METHOD: 100%

ACCURACY OF ADAPTIVE SKIPPING: 100%

TEST III:

Table 3. Results obtained in Test III

Input	Output	Output obtained by standard method	Output obtained by adaptive skipping
-50	-154	-154.00	-154.00
-100	-304	-304.00	-304.00
100	296	296.00	296.00
50	146	146.00	146.00
25	71	71.00	71.00

ACCURACY OF STANDARD METHOD: 100%

ACCURACY OF ADAPTIVE SKIPPING: 100%

Since this needs small training set, the reduced input samples are less. For the larger application the reduction are considerable since the skipping will increase as the number of epoch increases.

5. FUTURE WORK

The new algorithms are further worked to increment the value of *n* exponentially with out affecting the accuracy and test it for three types of applications that is applications requiring small training set, medium level and large set.

6. CONCLUSION

Thus the idea of adaptive stochastic sample presentation by **Adaptive Skipping** algorithm increases the learning rate an ANN, with no loss in accuracy. So the new algorithm can be used with any supervised learning algorithm to improve the learning rate of an ANN.

7. REFERENCES

[1] Owens, Aaron J., "Empirical Modeling of Very Large Data Sets Using Neural Networks", IJCNN 2000, vol. 6, pp. 6302-10.

[2] Schiffmann, W., Joost, M. and Werner, R., "Comparison of Optimized Backpropagation Algorithms", Artificial Neural Networks, European Symposium, Brussels, 1993.

[3] Schiffmann, W., Joost, M. and Werner, R., "Optimization of the Backpropagation Algorithm for Training Multilayer Perceptions", University of Koblenz: Institute of Physics, 1994.

[4] Zhang, Byoung-Tak, "Accelerated Learning by Active Example Selection", International Journal of Neural Systems, 5(1), Germany, 1994, pp. 6775-79.

[5] Riedmiller, Martin and Braun, Heinrich, "A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm", Proceedings of the IEEE Conference on Neural Networks, San Francisco, 1993.

[6] Hsin, Li, Sun, Sclabassi, R.J.;" A adaptive training algorithm for back-propagation neural networks",in proceedings of the IEEE transactions on tems, Man, and Cybernetics Society, Vol.25, No.3, pp.512-514, Aug. 2002.

[7] Russell, S. and P. Norvig, 2003. Artificial Intelligence: A Modern Approach. 2nd Edition. Prentice Hall, Inc.

[8] Hecht-Nielsen, R., "Theory of the Backpropagation Neural Network," in Proceedings of 1989 International Conference on Neural Networks, Washington D.C., Vol.I, pp.593-601, June 1989.

[9] R. Jacobs. "Increased rates of convergence through learning rate adaptation". Neural Networks, 1(4), 1988.

[10] Hush, D.R., Salas, J.M. "Improving the learning rate of back-propagation with the gradient reuse algorithm", in Proceedings of IEEE International Conference on Neural Networks, Vol.I, pp. 441 - 447, June 1989.