The relative amount of information contributed by learning and by pre-specification in a SRN trained to compute sameness.

Introduction

Generalization is a polysemous word in Cognitive Science. In particular generalization in ANN is statistical, whereas in some domains it is an all or none, algebraic concept (e.g. whether you generalize the right move in a polar question or not). Then a completely different product of some type of Neural Network, we analyze the algebraic notion of generalization in Neural Networks by studying sameness.

Sameness is a generalized version of logical equivalence; in some domains it is an all or none, algebraic concept (e.g. either you are a bachelor or not), whereas in some domains it is a statistical concept (e.g. you are a bachelor with likelihood of 0.95). In Neural Networks by studying sameness. I show that generalization is hard and that the difficulties cannot be reduced to learning outside the training space, although there is some form of limitations of induction present. I then turn to study simple, input unit networks.

1. Learning but not generalization of sameness in traditional SRNs

Mean square error and test set error during training: 100 input units, 2 hidden units, 0.1 learning rate. Initial 50 training and 50 test input pairs. The training error is quite stable as can be seen in the graph.

2. Training and generalization in simple cases

No generalization within the training

Part of the training and test sets and the actual output of the network in the test set is shown below. Although not every example is shown, every input example is shown at least once. The network has two output units and each output unit has a binary (0/1) output. The network has 1000 input units and 1000 output units. Hence, in a single training epoch, the network can learn 1000/1000 = 10% of the possible connections. Thus if the notion of training space is to be used it has to be defined precisely. Parameters: 0.1 learning rate. Train set 500 vectors, test set 500 vectors.

3. Analysis of simple cases using McCulloch-Pitts neurons

Two hidden units per input are necessary.

4. The information provided by the architecture and by the training set.

The relative amount of information in the structure

The relative amount of information in the structure

5. The cost of architecture: Different architectures generalize differently

Givens that the the input dimensions are independent in determining equivalence, we set up a network consisting of two independent layers, each layer having 9 input units. The inputs are pairs of vectors. The inputs are pairs of vectors.

Conclusions

A. The basic result is that generalizing sameness is difficult when training with a partial set of inputs.

B. This is troublesome if one considers that for k=2 n-dimensional vectors there are 2^n pairs. Thus, for k=2 and n=100 (which is not exaggerated in neural terms) 2^100 pairs should be tried. The amount of training would be prohibitive.

C. Also, the generalization is not guaranteed even within the training set (Marcus et al, 2001) at all when using a straightforward interpretation of the training set (2).

D. Theoretically it seems certain that, at least for the minimal number of hidden units of a SRN needs to be twice the number of inputs. Second, there are no networks having two hidden units in a single layer that can learn continuous sameness. This can be solved by using another hidden layer or more hidden units. This can explain part of our results in (1).

E. Networks with two hidden units cannot generalize freely when they are exposed to novel training data. Part of the difficulty lies in the number of possible networks that can compute the partial training set (4).

The simplest sequence (two time-steps) there are an order of magnitude more networks that can compute a 3-events training set than the full 4 items. We expected to find 1 to 10 in randomly selected networks to be able to generalize; this is not the case, so there are more restrictions stemming from a finite error surface of sameness that can be hindering generalization. This should be studied further.

F. The addition of more input units (5) naturally requires more hidden units and the possibility of crosswalk between sets that should be independent. Eliminating the crosswalk enhances generalization.

Overall, to set up a network that can learn sameness from a partial set of examples requires fixing several parameters.

References


NCTE: Number of vectors presented during training.

NTE: Number of vectors presented during testing.

S(0)=S0, I(t)=\begin{cases} 1 & \text{if } I(t)-I(t-1) \\
0 & \text{otherwise}
\end{cases}

T(t)=e^{-\frac{||I(t)-I(t-1)||}{g}}, t>0

6. Summary of results of training networks with one input

The table shows the number of networks that can compute temporal equivalence but fail to converge in some cases that reach a training error below 0.001 and 0.01.

The table shows the number of networks that can compute temporal equivalence but fail to converge in some cases that reach a training error below 0.001 and 0.01.

3. Analysis of simple cases using McCulloch-Pitts neurons

Two hidden units per input are necessary.

4. The information provided by the architecture and by the training set.

The relative amount of information in the structure

The relative amount of information in the structure

5. The cost of architecture: Different architectures generalize differently

Givens that the the input dimensions are independent in determining equivalence, we set up a network consisting of two independent layers, each layer having 9 input units. The inputs are pairs of vectors. The inputs are pairs of vectors.

Conclusions

A. The basic result is that generalizing sameness is difficult when training with a partial set of inputs (1).

B. This is troublesome if one considers that for k=2 n-dimensional vectors there are 2^n pairs. Thus, for k=2 and n=100 (which is not exaggerated in neural terms) 2^100 pairs should be tried. The amount of training would be prohibitive.

C. Also, the generalization is not guaranteed even within the training set (Marcus et al, 2001) at all when using a straightforward interpretation of the training set (2).

D. Theoretically it seems certain that, at least for the minimal number of hidden units of a SRN needs to be twice the number of inputs. Second, there are no networks having two hidden units in a single layer that can learn continuous sameness. This can be solved by using another hidden layer or more hidden units. This can explain part of our results in (1).

E. Networks with two hidden units cannot generalize freely when they are exposed to novel training data. Part of the difficulty lies in the number of possible networks that can compute the partial training set (4).

The simplest sequence (two time-steps) there are an order of magnitude more networks that can compute a 3-events training set than the full 4 items. We expected to find 1 to 10 in randomly selected networks to be able to generalize; this is not the case, so there are more restrictions stemming from a finite error surface of sameness that can be hindering generalization. This should be studied further.

F. The addition of more input units (5) naturally requires more hidden units and the possibility of crosswalk between sets that should be independent. Eliminating the crosswalk enhances generalization.

Overall, to set up a network that can learn sameness from a partial set of examples requires fixing several parameters.