Title
Shared Network Resources and Shared Task Properties

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 18(0)

Author
Munro, Paul W .

Publication Date
1996

Peer reviewed

Shared Network Resources and Shared Task Properties

Paul W. Munro
Department of Information Science
University of Pittsburgh
Pittsburgh PA 15260
munro@lis.pitt.edu

Introduction

The decomposition of a task as it is processed by the layers in a feed forward network depends on how the network is trained to the task. Analysis of hidden unit representations (using clustering techniques, for example) from networks trained by backpropagation has been used to demonstrate the contribution of individual layers to the full network computation. By training two networks to compute analogous tasks, similar computations might be expected in corresponding layers. This is not necessarily the case; however, if the two networks *share* weights at one of the layers, that shared layer is apt to compute the portion that is common to the tasks. This conjecture is demonstrated using a set of three analogous tasks, A, B, and C. A and B are simultaneously trained on two three-layer nets, where the middle layer of weights is shared. A third network is trained on task C, with the middle layer initialized to the weights learned for the first two tasks. Also, the middle (shared) layer was *not modified* during training on the target task (C) in the cases where A and/or B preceded it. The resulting increase in learning speed supports the conjecture that weights shared by networks computing different (but analogous tasks) come to compute those components of the tasks that are common to both.

Architecture and Tasks

Three networks were trained using backprop (Rumelhart, Hinton, and Williams, 1986) to illustrate the concept put forward in this paper (Figure 1). Networks A and B are trained on analogous tasks, sharing weights at the middle layer (of three layers). The shared layer is then held fixed during training of a third task on Network C. Network A is trained to give the coordinates of the fourth vertex of a square, given the other three, where is square is oriented rectilinearly with respect to the coordinate axes. Note that this task can be accomplished by copying certain input coordinates at the output layer. Network B is trained on the same task, with the squares rotated 45 degrees (in a "diamond" orientation). This task is slightly more complicated, but also easier than Task C which imposes no constraints on the orientation of the square (Figure 2). For the experiments reported here, 4 units were used in each of the hidden layers. The input values were in the range (0,1), both layers of hidden units used sigmoid units (that ranged from -1 to +1), and the output units were linear.

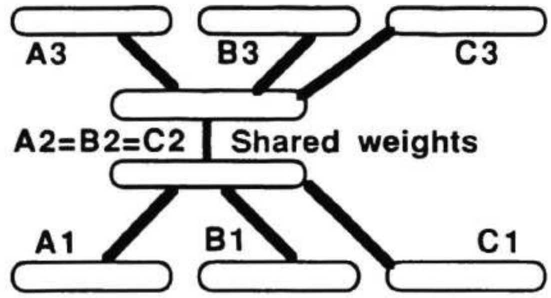

Figure 1. Architecture.

Figure 2. Tasks.

Results

These results, while preliminary, are illustrative. Each represents an average of 5 simulations with different starting weights.

Task C alone:	7200
Using weights from Task A:	3650
Using weights from Task B:	3600
Using weights from Task A and B together:	2950

Discussion

The results are supportive of the paper's conjecture that appropriate weight sharing may extract higher-order structural similarities among tasks. Of course, further work is required to demonstrate the idea convincingly

Reference

Rumelhart, D., Hinton, G. And Williams, R. (1986). Learning internal representations. In: Rumelhart, D., McClelland (Eds.), *Parallel Distributed Processing* Cambridge MA: MIT Press.