Information Sciences 294 (2015) 612–627

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Topology and simulations of a Hierarchical Markovian Radial Basis Function Neural Network classifier

Yiannis Kokkinos, Konstantinos G. Margaritis *

Parallel and Distributed Processing Laboratory, Department of Applied Informatics, University of Macedonia, 156 Egnatia str., P.O. Box 1591, 54006 Thessaloniki, Greece

A R T I C L E I N F O

Article history:
Received 28 December 2013
Received in revised form 31 July 2014
Accepted 6 August 2014
Available online 13 August 2014

Keywords:
RBF Neural Network
Hierarchical
Markov
Classification

A B S T R A C T

This paper presents a Hierarchical Markovian Radial Basis Function Neural Network (HiMarkovRBFNN) model that enables recursive operations. The hierarchical structure of this network is composed of recursively nested RBF Neural Networks with arbitrary levels of hierarchy. All hidden neurons in the hierarchy levels are composed of truly RBF Neural Networks with two weight matrices, for the RBF centers and the linear output weights, in contrast to the simple summation neurons with only linear weighted combinations which are usually encountered in ensemble models and cascading networks. Thus the neural network operation in every node is exactly the same at all levels of the hierarchical integration. The hidden RBF response units are recursive. The training methods also remain the same for all levels as in a typical single RBF Neural Network. The simplicity in the neural network construction process is demonstrated by means of three textbook algorithms, namely the well known *k*-means clustering, the classical tree-based recursion function and the standard regularized least squares solver. Determining centers can be performed top-down and calculation of linear output weights is performed bottom-up. The framework is rather general and optimization algorithms can also be applied. Experimental simulations on various benchmark datasets show that the recursive operation for the hidden RBF response units is promising. Comparisons with the two standard model meta-learning architectures, namely Committee Machines and Cascaded Machines, reveal that the proposed method produces similar results and compares well as a combiner that can merge many HiMarkovRBFNN child nodes into one higher level parent HiMarkovRBFNN node of the same functionality.

1. Introduction

In parallel processing [39] hierarchical structures intrinsically facilitate parallelism, and naturally alleviate many common scalability problems through the divide-and-conquer strategy [12]. For example in order to manage a large and complex dataset we can divide the data so as to create a structure of hierarchical nested data clusters. Similarly, a hierarchical neural network [29] tries to break up a complex task into a series of simpler and faster computations at each level of the hierarchy. When the neural network is in operation, the outputs of the lower levels are combined into the higher levels. Scalability issues can be alleviated, if the problem in question can be divided and solved in several independent sub-tasks [12]. In this way computational resources can be economized when the neuron connections break up into smaller hierarchical groups

* Corresponding author.

http://dx.doi.org/10.1016/j.ins.2014.08.025

that comprise modules with fewer sub-connections. During training the hierarchical decomposition inherently facilitates parallelism. Overall computation can be faster when the simpler lower level operations can be performed independently in parallel. Although hierarchies offer a major advantage for efficient implementations there are fundamental issues concerning sample complexity, functional decomposition, hierarchical representation, and training algorithms.

Thus, the main target of this study is to build a hierarchical modular nested RBF Neural Network classifier which has levels of training that make possible a faster hierarchical learning. The RBF units are organized in a hierarchical fashion and the network adapts itself into a structure of hierarchical nested data clusters. We demonstrate the simplicity in the building process by using three textbook algorithms, namely the well-known *k*-means clustering, the classical tree-based recursion function for the hidden response units and the standard regularized least squares solver for the linear output weights.

The Radial Basis Function Neural Network (RBFNN) [4,5,14,20,23,25,31,33,34,38] is one of the most representative types of Artificial Neural Network models, well established over the years, and widely applied in many science and engineering fields. Owning to their compact topology and fast learning RBFNNs have been extensively used for function approximation, classification, system identification and control tasks. At present, the many recent studies [1,2,15,22,30,42] on RBFNNs and their variations prove the continuing interest on them. A single RBFNN uses Radial Basis Functions (RBF) as local activation units. This approach was inspired by the presence of many locally response units that exist in the human brain. Typically Gaussian functions are employed as hidden RBF units. Those RBFs can facilitate hierarchical learning.

In view of the fact that the locality principle operation of RBF units implies that a hierarchy is feasible some very interesting hierarchical RBF Neural Network models have appeared during the last years. Pioneering works include the hierarchical RBF model of Ferrari and co-workers [6,16–18] that was specialized for multi-scale approximation in 3-D meshing. The hierarchical RBF model of Mat Isa et al. [28] was proposed for clinical diagnosis. The hierarchical RBF model of Van Ha [37] that employs the k-nearest rule and several stage networks was proposed to solve medical diagnosis problems. The flexible hierarchical RBF model by Chen and co-workers [8–10] was applied in system identification, classification and time-series forecasting. Although not proposed for RBFNN we must mention the well-known hierarchical mixture of experts of Jordan and Jacobs [24] suitable for hierarchical mixing experts with gating networks. The existing hierarchical models are applied for different purposes and use different training algorithms than those in the original RBFNN since the first is a multi-gridding approach, the second is a cascading approach of two neural network modules, the third is a cascading of several modules, the fourth is an evolving neural tree and the fifth is a mixture of experts. However, their common ground is that they all built the prediction function through some form of hierarchical iterative approximation.

Several types of hierarchies in the previous models can be recognized. At this point we attempt a primary categorization of the hierarchical types, in order to create a small taxonomy and use a common terminology. For example, multi-resolution is the use of multiple scales of resolution. Modular usually refers to a system composed of many neural network modules. Cascading typically means that the outputs of the first level are used as inputs into the second level in the hierarchy. In the model of Ferrari and co-workers there are many RBF units, with different in scale width parameters, aligned in a 3D grid multi-resolution hierarchy and act as tree nodes and one RBFNN that composed of all of them. Ferrari et al. uses a linear combining function, the weighted sum of all RBFs. Mat Isa et al. uses two RBFNN modules cascading together in which the outputs of the first become inputs of the second. In the model of Chen and co-workers there are many small RBFNNs arranged in a hierarchy to form an acyclic graph, whose RBFNN nodes have different input features and are composed of as many hidden RBF neurons as their input neurons. Chen et al. model also uses cascading, in which outputs of the previous level RBFNNs become inputs of the next. Within this context one can detect many types of hierarchies, like hierarchical multi-resolution (Ferrari et al. model), two-level modular cascading (Mat Isa et al. model), hierarchical modular cascading (Chen et al. model, Van Ha model), hierarchical mixture of experts (Jordan and Jacobs model) and hierarchical modular nested (this work). The last type, which is the subject of our study, has many RBFNN modules hierarchically nested together with a recursive operation. Section 2 will further elaborate on the existing types of hierarchies.

The contribution of the paper and the essence of the proposed solution are the following. For a classification function one can observe that the main problem for an intrinsically hierarchical structure is to support recursion, meaning same operation at all levels. Without defining and using recursion it is difficult to build a truly hierarchical nested structure with the same functionality from top to bottom. In view of this problem we employ the Markov rule in order to define a recursive RBF response function. Accordingly, we proposed a Hierarchical Markovian RBF Neural Network (HiMarkov RBFNN) topology and a training framework that can use the same training stages and algorithms from the conventional RBFNNs. In the present article (which elaborates and improves on our earlier work [26]) the HiMarkovRBFNN has a tree structure with a clear recursive operation in every node. Generally, in a tree structure of data one can take many close to each other data clusters and merge them into a single higher level cluster that contains all of them nested. In the same way, a parent HiMarkovRBFNN node takes many children HiMarkovRBFNN nodes and merges them into a single higher level node that contains all of them nested. Thus a distinctive feature of the proposed approach is that each hidden 'neuron' in every level of the hierarchy is another fully-functional HiMarkovRBFNN having inside other HiMarkovRBFNNs nested, and not just simple units that combine the previous level outputs. The classical two synaptic weight sets, for the RBF centers and the linear output weights, are present in each module and the topology is the same for all. The operation of all the modular nodes starting from the bottom level in the hierarchy and moving toward the top is also the same, as a result of the recursion. In addition, all those HiMarkovRBFNNs are individually trained in the same way, using conventional RBFNN training algorithms, well known from the literature. The selection of the hierarchically arranged RBF centers can be performed top-down from coarse-grained to