Neurocomputing 125 (2014) 7–16

Contents lists available at ScienceDirect

Neurocomputing

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

An online self-adaptive modular neural network for time-varying systems

Junfei Qiao [a,*], Zhaozhao Zhang [a,b], Yingchun Bo [a,c]

[a] College of Electronic and Control Engineering, Beijing University of University of Technology, Beijing, 100124, China
[b] Institute of Electronic and Information Engineering, LiaoNing Technical University, Liaoning, 125105, China
[c] College of information and Control Engineering, China University of Petroleum, Shandong, 266555, China

ARTICLE INFO

Article history:
Received 16 April 2012
Received in revised form
28 August 2012
Accepted 13 September 2012
Available online 1 March 2013

Keywords:
Modular neural network
Self-adaptive
Time-varying system
Fuzzy strategy
Online

ABSTRACT

We propose an online self-adaptive modular neural network (OSAMNN) for time-varying systems. Starting with zero subnetworks, OSAMNN uses a single-pass subtractive cluster algorithm to update the centers of radial-basis function (RBF) neurons for learning. Then the input space can be partitioned. The OSAMNN structure is capable of growing or merging subnetworks to maintain suitable model complexity, and the centers of RBF neurons can also be dynamically adjusted according to changes in the data environment. A fuzzy strategy is applied to select suitable subnetworks to learn the current sample. This method yields improved learning efficiency and accuracy. OSAMNN can adapt its architecture to realize online modeling of time-varying nonlinear input–output maps. Results for experiments on benchmark and real-world time-varying systems support the proposed techniques.

1. Introduction

Feedforward neural networks are widely used in many fields because of their ability to directly approximate complex nonlinear mappings from input–output samples [1]. Various types of neural network and learning algorithm have been developed to solve different kinds of practical problems [2–6]. An increasing number of real life problems are complex time-varying environments, such as a plant process that undergoes seasonal variations in its input–output behavior. However, feedforward neural networks are fully coupled networks in which each input activates all hidden nodes. This characteristic not only leads to a large calculation cost but also makes knowledge accumulation extremely difficult, since the networks tend to forget previously learned mappings rather quickly when exposed to new mappings [7].

Modular design is an effective method for alleviating this forgetfulness and decreasing the computation costs common for fully coupled neural networks [8–10]. The concept of modularity in artificial neural networks was inspired by biological and psychological research proving that modularity is key to the efficiency and intelligence of the human brain [11]. Modular neural networks (MNNs) adopt the principle of “divide and conquer”, which means that each MNN system consists of several specialized modules, such as feedforward neural network models. Each of the modules performs a subtask of the original complex task, each has a simpler structure, and each evaluates the same or distinct input samples without communicating with other modules. After each module obtains its independent results, an integration unit combines the results in a predefined method to generate the overall output of the complex system [12].

Because there are many advantages to adopting a modular approach [11], MNNs are widely used in many fields. Molina-Vilalana et al. proposed an MNN for learning of grasping tasks [13]. Gradojevic et al. applied a nonparametric MNN to price S&P-500 European call options [14]. Tseng and Almogahed aggregated MNNs using a genetic algorithm and applied this methodology to profiling of an aircraft trajectory pattern [15]. However, these fields are very different from time-varying systems because the training data were known in advance. For time-varying systems, especially more complex time-varying data sets that exhibit regime-shifting properties, the training data are presented to the MNN one by one, and the MNN needs to acquire knowledge by continuous self-organization of its all parameters to suit changing patterns in evolving data streams [16]. This strong distinction in learning objectives is essential for the MNN to model data environments with changing characteristics. Therefore, to model time-varying systems, an MNN should be able to adapt all the parameters online and respond well to localized changes in the input–output map by adjusting only a few parameters. The MNN should also be able to dynamically adjust its complexity by pruning or growing to

* Corresponding author. Tel./fax: +86 10 67391459.
E-mail address: isibox@sina.com (J. Qiao).

0925-2312/$ - see front matter
http://dx.doi.org/10.1016/j.neucom.2012.09.038

properly match the complexity of the current problem being modeled [17]. The structures of MNNs that exhibit some of these desired characteristics have been described [17–19]. Ramamurti and Ghosh proposed a structurally adaptive MNN (SAMNN) for nonstationary environments [17]. This neural network is capable of dynamically adapting its architecture online by growing or pruning subnetworks for modeling of time-varying nonlinear input–output maps. However, growth or pruning of SAMNN is implemented at a fixed step in the training, without considering the dynamic shifting properties of time-varying systems. Chen et al. proposed a self-generated MNN architecture for supervised learning in which a tree-structured MNN is automatically generated with a growing algorithm during learning [18]. However, it does not include a pruning process. Loo et al. proposed a self-regulating growing multiexpert network structure (SGMNN) in which growth and pruning are also implemented at a fixed step in the training [19]. In addition, all of these approaches use an error-driven method to establish the MNN structure; for example, the criterion to determine growth or pruning of subnetworks is based on a mean-squared error rule, which leads to slow MNN growth.

To overcome these problems, we propose an online self-adaptive modular neural network (OSAMNN) for time-varying systems that is motivated by localized modeling [20] and multiple model approaches [21]. Starting with zero subnetworks, OSAMNN adopts a single-pass learning approach in which an improved subtractive clustering algorithm is applied to identify the system operating conditions. This clustering algorithm has been successfully used for the design of fuzzy neural network structures [22,23]. Feedforward neural networks are used to establish a number of local models in different operating conditions. Because each cluster center (the centers of RBF neurons; Fig. 1) corresponds directly to a subnetwork and the cluster centers can be dynamically created, merged or shifted according to the time-varying environment, the OSAMNN structure can adaptively grow or merge subnetworks to maintain appropriate complexity at all times. In the OSAMNN learning process, a fuzzy strategy is applied to select suitable subnetworks to train the current sample. This method leads to improved learning efficiency and accuracy.

There are some new characteristics in the proposed OSAMNN for time-varying systems. (1) Unlike most MNNs, OSAMNN uses a method driven by the input data stream to establish its structure. The advantage of this method is rapid establishment of the OSAMNN structure at the start of learning. (2) A fuzzy strategy is applied to select suitable subnetworks to learn the current input data, which improves the learning accuracy and efficiency. (3) OSAMNN can start with zero subnetworks and only one data sample. These interesting features make OSAMNN potentially very useful in real-time systems and as a tool for modeling time-varying systems. We applied OSAMNN to a Mackey–Glass chaotic time series, identification of a nonlinear dynamic system, and prediction of water quality for wastewater treatment. Experimental results confirm the viability and efficiency of our technique.

The remainder of the paper is organized as follows: Section 2 describes the OSAMNN structure. Section 3 discusses the online identification of cluster centers. Section 4 introduces the method for dynamic integration of subnetworks using a fuzzy strategy. Section 5 describes the experimental analysis and performance benchmarks. Section 6 summarizes and concludes.

2. OSAMNN structure

OSAMNN is an online neural connectionist construct with five layers. It is a multi-input, single-output system (Fig. 1). Each layer performs specific operations in tandem to realize an online learning task. The five layers are the input layer, the RBF layer, the task allocation layer, the subnetwork layer, and the integration output layer. The role and mathematical description of each layer in OSAMNN are as follows.

2.1. Layer 1

The input layer consists of n source nodes, where n is the dimensionality of the input variables at time k:

$$\mathbf{x}(k) = [x_1(k), \cdots, x_n(k)]^T. \tag{1}$$

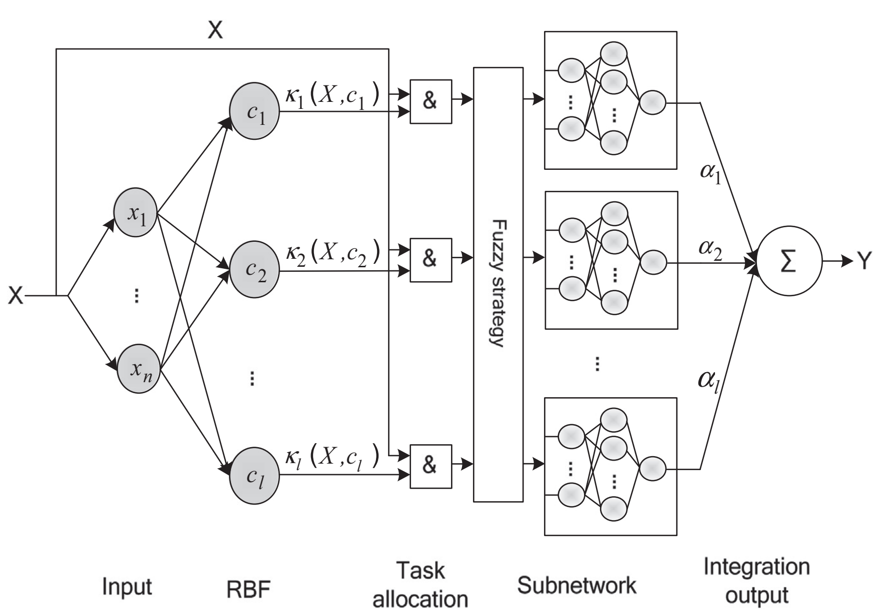

Fig. 1. Structure of OSAMNN.