ARTICLE IN PRESS

Knowledge-Based Systems xxx (2015) xxx–xxx

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Optimizing the echo state network with a binary particle swarm optimization algorithm

Heshan Wang, Xuefeng Yan *

Key Laboratory of Advanced Control and Optimization for Chemical Processes of Ministry of Education, East China University of Science and Technology, Shanghai 200237, PR China

ARTICLE INFO

Article history:
Received 10 November 2014
Received in revised form 2 June 2015
Accepted 6 June 2015
Available online xxxx

Keywords:
Echo state network
Reservoir computing
Artificial neural network
Binary particle swarm optimization algorithm
Time-series prediction

ABSTRACT

The echo state network (ESN) is a novel and powerful method for the temporal processing of recurrent neural networks. It has tremendous potential for solving a variety of problems, especially real-valued, time-series modeling tasks. However, its complicated topologies and random reservoirs are difficult to implement in practice. For instance, the reservoir must be large enough to capture all data features given that the reservoir is generated randomly. To reduce network complexity and to improve generalization ability, we present a novel optimized ESN (O-ESN) based on binary particle swarm optimization (BPSO). Because the optimization of output weights connection structures is a feature selection problem and PSO has been used as a promising method for feature selection problems, BPSO is employed to determine the optimal connection structures for output weights in the O-ESN. First, we establish and train an ESN with sufficient internal units using training data. The connection structure of output weights, i.e., connection or disconnection, is then optimized through BPSO with validation data. Finally, the performance of the O-ESN is evaluated through test data. This performance is demonstrated in three different types of problems, namely, a system identification and two time-series benchmark tasks. Results show that the O-ESN outperforms the classical feature selection method, least angle regression (LAR) method in that its architecture is simpler than that of LAR.

1. Introduction

Recently, reservoir computing (RC) [1,2] has drawn much attention from the machine learning community as a novel recurrent neural network (RNN). RC differs from other algorithms in two important ways: first, a large and untrained dynamic reservoir is used. Second, the desired output function is usually implemented through a linear memory-less mapping of the full instantaneous state of a dynamical system. Some examples of popular RC methods are echo state networks (ESNs) [3–5], liquid state machines [6], back-propagation decorrelation neural networks [7] and evolution of recurrent systems with linear outputs [8]. In this study, we focus on the ESN approach, which is among the simplest effective RC forms.

ESN is characterized by the use of an RNN with a fixed untrained reservoir and a simple linear readout. This reservoir contains a large number of randomly and sparsely connected neurons. The sole trainable component is the readout weights, which can be obtained through simple linear regression. ESNs have been applied successfully to a wide range of real-world domains, including time-series prediction [9,10], batch bioprocesses modeling [11], nonlinear dynamic system identification [12,13], speech processing [14], mobile traffic forecasting [15], gas turbine prediction [16], stock price prediction [17] and language modeling [18]. However, ESN is occasionally criticized for its blackbox nature: the reservoir connectivity and weight structure are generated randomly beforehand, thus, the process of establishing optimal reservoirs for a given task is an issue [19]. An ESN transforms an incoming time-series signal into a high-dimensional state space. Not all dimensions may contribute to the solution. The internal layer of ESN is sparsely connected, hence, the fact that each output node is connected to all internal nodes seems contradictory [20]. Therefore, the output connection of ESN should be optimized.

Many researchers have recently focused their efforts on new ways to optimize the architecture of artificial neural networks, including the methods of pruning [21,22], construction [23] and evolutionary algorithms [24–26]. Constructing algorithms start training with a small network and incrementally add hidden nodes during training when the network cannot reduce the training error. However, there are also some issues that constructive algorithms

* Corresponding author at: P.O. Box 293, MeiLong Road No. 130, Shanghai 200237, PR China. Tel./fax: +86 21 64251036.

E-mail address: xfyan@ecust.edu.cn (X. Yan).

http://dx.doi.org/10.1016/j.knosys.2015.06.003
0950-7051/

Please cite this article in press as: H. Wang, X. Yan, Optimizing the echo state network with a binary particle swarm optimization algorithm, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.06.003

need to overcome. For example, when a network is grown, there is no guarantee that all of the added hidden nodes are properly trained. Pruning algorithms start with an over-sized network and remove unnecessary network parameters, either during training or after convergence to a local minimum. Unfortunately, one of the disadvantages of pruning algorithms is their heavy computational burden since the majority of the training time is spent on networks which are larger than necessary. Evolutionary learning algorithms have shown great capability to solve problem of feed-forward neural networks recently. In [27], SeyedAli et al. proposed a hybrid particle swarm optimization and gravitational search algorithm to train feed-forward neural networks in order to reduce the problems of trapping in local minima and the slow convergence rate of current evolutionary learning algorithms.

Dutoit et al. treat optimization of output weights connection structures as a feature selection problem and proposed several classical feature selection methods such as, all subsets, forward selection, backward elimination and least angle regression (LAR) to investigate how pruning some connections from the reservoir to the output layer can help increase the generalization capability of reservoir computing. [28]. Kobialka and Kayani use a greedy feature selection algorithm to exclude irrelevant internal ESN states [20]. The optimization of the connection structure of output weights is a problem of whether internal and output layer nodes are connected, thus, it is a discrete optimization problem. Moreover, the dimension of optimization variable is equal to the number of internal neurons. The number of internal neurons (generally 200–1000 neurons) must be large enough to capture all data features given that the reservoir is generated at random. In sum, the optimization of the connection structure of output weights of ESN is a discrete, high-dimension, complex, and strongly nonlinearity feature selection problem. Existing feature selection approaches, such as greedy search algorithms, suffer from a variety of problems, such as stagnation in local optima and high computational cost [29]. Therefore, an efficient global search technique is needed to address feature selection problems. Many studies report that evolutionary computation algorithm effectively solve such computational problems. Evolutionary computation techniques are well-known for their global search ability, and have been applied to feature selection problems. These includes particle swarm optimization (PSO) [30,31] and genetic algorithms (GAs) [32]. Compared with GAs, PSO is easier to implement, has fewer parameters, computationally less expensive, and can converge more quickly [33]. Due to these advantages, PSO has been used as a promising method for feature selection problems. PSO is such a global search technique, which is computationally less expensive, easier to implement, has fewer parameters and can converge more quickly than other techniques, such as GA.

PSO is a population-based optimization technique that emulates the social behavior of animals, such as the swarming of insects, the flocking of birds, and the schooling of fish, when searching for food in a collaborative manner. This technique was originally designed and introduced by Eberhart and Kennedy [34,35] and has been widely applied in fields such as optimal control and design [36,37], biomedical [38–40], clustering and classification [41,42], electronics and electromagnetics [43,44], bi-level pricing problems in supply chains [45], and modeling [46–48]. The original PSO has been utilized in continuous space, where in trajectories are defined as changes in position on an number of dimensions. However, many optimization problems occur in discrete space. Thus, the traditional PSO cannot solve binary combinational optimization problem, such as structural topology optimization. Thus, [49] introduce a discrete binary version of PSO that can be used on discrete binary variables. In continuous PSO, trajectories are defined as changes in position on an number of dimensions. By contrast, binary particle swarm optimization (BPSO) trajectories are changes in the probability that a coordinate will take on a value of zero or one. BPSO has been used in many applications, such as the problem of instance selection for time series classification [50], ear detection [51], and feature selection [52].

There are some flaws still exist in original BPSO such as local minima and slow convergence speed. [53] proposed a binary version of bat algorithm to improve BPSO and the results prove that the proposed binary bat algorithm is able to significantly improve the performance on majority of the benchmark functions. [54] proposed a hybrid PSO and gravitational search algorithm (GSA) and the results prove that this hybrid algorithm outperforms both PSO and GSA in terms of improved exploration and exploitation.

Ref. [55] presented a new discrete BPSO that addresses the difficulties associated with the original BPSO. To improve the generalization performance of ESN and to simplify its structure, the current paper introduces an optimized ESN (O-ESN) that utilized the new BPSO algorithm [55] to optimized the structure of connections from reservoir to the output layer of ESN. The remainder of this article is organized as follows: Section 2 provides a brief overview of ESN design and training. Section 3 presents a short review of the BPSO algorithm. Section 4 discusses the experimental results. Finally, Section 5 presents a brief conclusion.

2. Echo state network

2.1. Architecture of the ESN

An ESN is composed of three parts, as illustrated in Fig. 1: The left component has K input neurons, the internal (reservoir) part has N reservoir neurons, and the right component has L output neurons. The reservoir state $\boldsymbol{s}(t)$ and output $\boldsymbol{o}(t)$ at discrete time step t are described by [56]:

$$\boldsymbol{s}(t) = f(\boldsymbol{W}_u^s \cdot \boldsymbol{u}(t) + \boldsymbol{W}_s^s \cdot \boldsymbol{s}(t-1) + \boldsymbol{W}_o^s \cdot \boldsymbol{o}^T(t-1)) \quad (1)$$

$$\boldsymbol{o}(t) = \boldsymbol{s}^T(t) \cdot \boldsymbol{W}_s^o \quad (2)$$

where f is the reservoir activation function (typically a hyperbolic tangent or another sigmoidal function). $\boldsymbol{u}(t), \boldsymbol{s}(t)$ and $\boldsymbol{o}(t)$ are the input, reservoir state, and output at discrete time step t respectively. The connection weights between the input neurons and the reservoir are presented in a $N \times K$ weight matrix $\boldsymbol{W}_u^s$. The

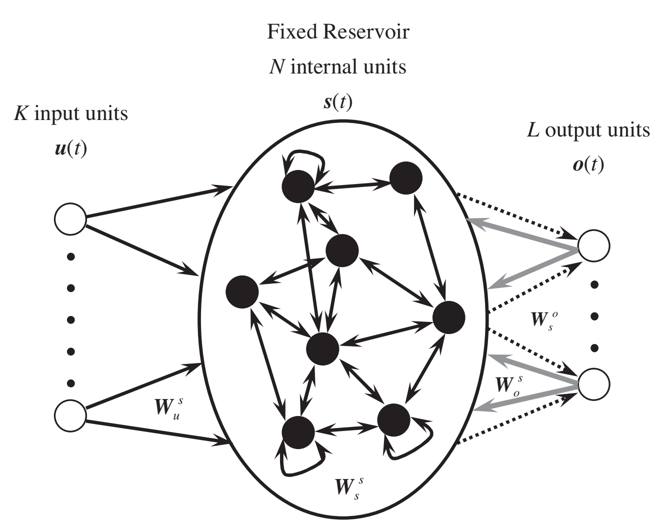

Fig. 1. The basic architecture of ESN. Dashed arrows indicate connections which are trained in the ESN approach. Shaded arrows indicate feedback connections that are possible but not required. Black solid arrows indicate connections which are random created and fixed during training.