Neurocomputing 174 (2016) 322–330

Contents lists available at ScienceDirect

Neurocomputing

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

Extreme learning machine for time sequence classification

Huaping Liu [a,*], Lianzhi Yu [b], Wen Wang [b], Fuchun Sun [a]

[a] Department of Computer Science and Technology, Tsinghua Unviersity, State Key Lab. of Intelligent Technology and Systems, TNLIST, Beijing, P.R. China
[b] School of Optical-Electrical and Computer Engineering, University of Shanghai for Science & Technology, P.R. China

ARTICLE INFO

Article history:
Received 23 August 2014
Received in revised form
28 November 2014
Accepted 20 January 2015
Available online 2 September 2015

Keywords:
Extreme learning machine
Time sequence classification
Linear dynamical systems
Bag-of-systems

ABSTRACT

In this paper, a new framework to effectively classify the time sequence is developed. The whole time sequence is divided into several smaller sub-sequence by means of the sliding time window technique. The sub-sequence is modeled as a linear dynamic model by appropriate dimension reduction and the whole time sequence is represented as a bag-of-systems model. Such a model is very flexible to describe time sequence originated from different sensor source. To construct the bag-of-systems model, we design the codebook by using the K-medoids clustering algorithm and Martin distance between linear dynamic systems. Such a technology avoids the problem that linear dynamic systems lie in non-Euclidean manifold. After obtaining the represented of time sequence, an extreme learning machine is utilized for classification. Finally, the proposed method is verified on some benchmark and shows that it obtains promising results.

1. Introduction

Time sequence is ubiquitous in many fields. For instance, the human–robot-interface may require to classify the gaits, gestures, or actions, all of which are representative time sequences. Especially, human activity recognition has become an important emerging field of research within context-aware systems [4,5]. Reference [6] presented a wearable activity sensor system and a systematic activity classification scheme for the classification of human daily physical activities. The wearable activity sensor system, consisting of two activity sensor modules worn on dominant hand wrists and ankles of the users, is used for collecting activity acceleration signals. Other similar studies focused on how one can use a variety of accelerometers to identify a range of user activities.

The Dynamic Time Warping (DTW) distance has been extensively utilizedfor time sequence classification. It allows a measure of the similarity invariant to certain nonlinear variations in the time dimension and attempts to compensate for possible time translations/dilations between patterns. However, for long sequence, it is more appropriate to measure similarity from higher level structure but not point-to-point local comparisons. In [2], a Bag-Of-Features (BoF) approach in which complex objects are characterized by feature vectors of subobjects is proposed to tackle the problem of time sequence classification. The BoF representation allows one to integrate local information from segments of the time series in an efficient way. But this work is still based on the shape-based features such as the slope and variance. In [31], the Linear Dynamic Systems (LDS) model is used to construct a Bag-of-Systems (BoS) framework to classify visual dynamic texture. LDS is a powerful tool to model the rich time sequence. In [11] it was used to model the visual dynamic texture, and in a recent literature [22] the authors used such a model to discuss the intrinsic relation between control and machine learning. All of the above-mentioned methods transform the original time sequence into histogram representation and use the popular Support Vector Machine (SVM) to design the classifier.

On the other hand, Extreme Learning Machine (ELM) [14,15] has attracted more and more researchers' attention for its better performance than traditional parameters learning algorithm such as gradient descent algorithm in generalized single hidden layer feed-forward neural networks (SFLNs). In [16], the authors have proved that ELM tends to have better scalability and achieve similar (for regression and binary class cases) or much better (for multi-class cases) generalization performance at much faster learning speed (up to thousands times) than traditional SVM. ELM has been used in several domains ranging from human action recognition [17,8,25], face recognition [32,26], visual tracking [20] and so on.

Motivated by the advantage of ELM and LDS, we regard time sequences as the output of an intrinsic dynamic system shown in Fig. 1. To obtain more complete representation for the time sequence, we use un-ordered multiple local LDSs to represent the whole time series. As soon as the features of each time series are

* Corresponding author.
E-mail address: hpliu@tsinghua.edu.cn (H. Liu).

http://dx.doi.org/10.1016/j.neucom.2015.01.093
0925-2312/

obtained, we can train a classifier for recognition. The main contributions are summarized as follows:

1. The whole time sequence is divided into several smaller sub-sequence by means of the sliding time window technique. The sub-sequence is reasonably modeled as LDS by appropriate dimension reduction. Further, the whole time sequence is represented as a BoS, which is a bag of LDS patches. Such a model is very flexible to describe time sequence originated from different sensor sources.
2. To model the BoS, a codebook is proposed, which utilizes the Martin distance between LDSs and avoid the problem that LDS lies in non-Euclidean manifold.
3. The obtained feature vector of time sequence is classified by an ELM, which provides strong generality and parameter insensitivity.

The rest of this paper is organized as follows. In Section 2, the overall architecture is illustrated. Section 3 reviews LDS and the metric for LDSs. In Section 4 we classify time sequences using proposed framework. Section 5 provides some experimental results. Finally, the conclusion is given in Section 6.

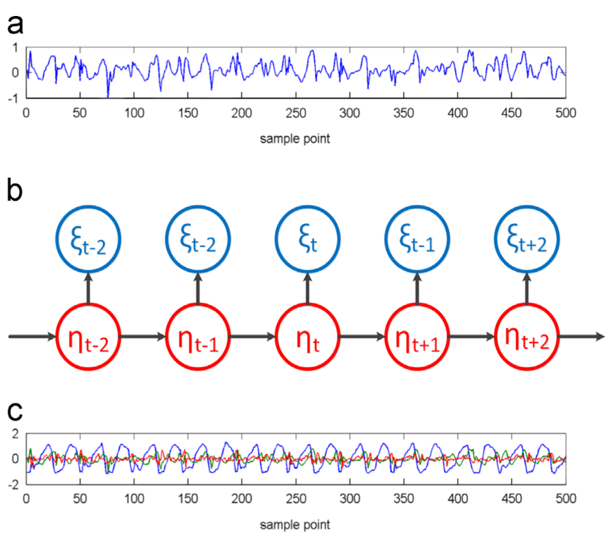

Fig. 1. Modeling time sequence as a dynamic system: (a) An waveform and it can be regarded as the output of dynamic system. (b) A simplified dynamic system. (c) The hidden state space of the dynamic system.

2. Architecture

The framework for time sequence recognition is inspired by the BoF approach to classify time series [1] and the bag of dynamical systems [27] in categorizing dynamic textures. The main steps in our framework are as follows:

1. Extract LDS models from the training set.
2. Form codebook using K-medoids clustering algorithm.
3. Represent time sequence using the formed codebook.
4. Train ELM using the representation vectors and corresponding labels.
5. Given a new time sequence, infer which class it belongs to using the trained ELM.

The developed framework is illustrated in Fig. 2. In the training stage, subsequences (*red rectangle*) are sampled from each time sequence and LDS models are extracted from the subsequences. Models are then grouped into K groups and the center of each group is selected to form the codebook. Once the codebook is formed, all of the time sequences can be represented by the codebook. ELM is then trained by the representation vectors and corresponding labels. After ELM is trained, the recognition can be performed. Given a new time sequence, the LDS are extracted by the same method in the training progress, and we can represent the time sequence with the formed codebook. Finally, we can infer which category the time series belongs to.

3. Brief review about LDS

3.1. LDS representation for time series

Assume that a time series $\{\xi_t\}_{t=1,\cdots,\tau}$, $\xi_t \in \Re^m$ is a realization of a second-order stationary stochastic process [11]. This means that the joint statistics between two time instances is shift-invariant. In our paper, we assume that there exists symmetric positive-definite matrices $\mathbf{Q} \in \Re^{n\times n}$ and $\mathbf{R} \in \Re^{m\times m}$ such that

$$\begin{cases} \eta_{t+1} = \mathbf{F}\eta_t + \nu_t & \nu_t \sim \mathbf{N}(0, \mathbf{Q}) \\ \xi_t = \mathbf{G}\eta_t + \omega_t & \omega_t \sim \mathbf{N}(0, \mathbf{R}) \end{cases} \tag{1}$$

where $\eta_t \in \Re^n$ is the hidden state at time t with initial condition η_0, $\mathbf{F} \in \Re^{n\times n}$ models the dynamics of the hidden state, $\mathbf{G} \in \Re^{m\times n}$ maps the hidden state to the output of the system, ν_t and ω_t are driven by Gaussian white noise.

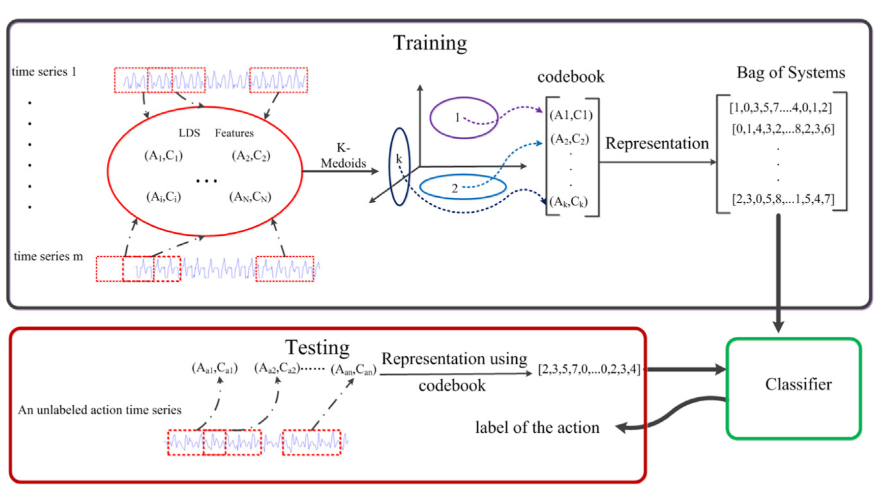

Fig. 2. Architecture of our framework. Top: Training progress. Bottom: Testing progress. Bottom right corner: ELM classifier.