Engineering Applications of Artificial Intelligence 57 (2017) 153–159

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

A regulated boosting technique for material fatigue property prognostics

De Z. Li[a], Wilson Wang[b,*], Fathy Ismail[c]

[a] eMech Systems Inc., Thunder Bay, ON, Canada P7B 7C1
[b] Department of Mechanical Engineering, Lakehead University, Thunder Bay, ON, Canada P7B 5E1
[c] Department of Mechanical & Mechatronics Engineering, University of Waterloo, Waterloo, ON, Canada N2L 3G1

ARTICLE INFO

Keywords:
Material fatigue property prediction
Boosting technique
Ensemble learning
Overfitting
Time series forecast

ABSTRACT

A new regulated boosting technique, rBoost, is proposed in this work for time series forecast and material fatigue property prognosis. The rBoost employs the principle of ensemble learning, associated with base predictors. Different from general boosting techniques that are prone to overfitting when using relatively strong base predictors such as autoregressive model and radial basis function, the proposed rBoost technique aims to improve error convergence and reduce the overfitting problem using the new sample weight regulator. The effectiveness of the developed rBoost predictor is firstly demonstrated by simulation tests, and then the rBoost is implemented for material fatigue property prognosis. Test results show that the proposed rBoost predictor is an effective forecast tool; it can capture system dynamics effectively and track system characteristics accurately.

1. Introduction

Time series forecast is a process that extrapolates future states of a dynamic system by analyzing its available (i.e., current and past) states. Reliable forecast information is valuable in many real-world applications such as electric load forecasting (Elattar et al., 2010; Bashir and EI-Hawary, 2009), financial indicator prediction (Lu, 2010; Chang and Liu, 2008), and equipment health condition monitoring (Wang, 2008). Fatigue induced damage is one of the most uncertain and highly unpredictable failure mechanisms for a large variety of mechanical and structural systems subjected to cyclic and random loads during their service life. Fatigue life prediction of materials is an on-going and unsolved challenge for non-destructive evaluation and structural condition monitoring. Material fatigue prognosis information could be used to estimate the material property propagation and remaining useful life (Peng et al., 2015; Lee et al., 2015; Corbetta et al., 2015). The classical time series forecast is mainly based on the use of analytical tools, such as autoregressive (AR) and autoregressive-moving-average (ARMA) models (Brockwell and Davis, 2009). However, the prediction efficiency of these analytical model-based methods depends on the accuracy of the formulated models. The alternative is the use of soft-computing tools such as neural networks (NNs) (Mohammadi et al., 2014; Khashei and Bijari, 2012) and neuro-fuzzy (NF) systems (Wang et al., 2015; Li et al., 2013). These soft computing techniques can be trained to recognize data characteristics in the time series. Their performance, however, may be limited due to suboptimal system structures and inefficient system training.

Another promising approach for system state forecasting is based on the use of the boosting technique. It is an ensemble learning approach that applies a series of weak learners whereby each weak learner deals with one tweaked data property (e.g., the data distribution) (Yu et al., 2016; Gao et al., 2012). For example, an AdaBoost.RT algorithm is presented in (Shrestha and Solomatine, 2006) to detect incorrect predictions and update data distribution accordingly. A gradient boosting technique is suggested in (Taieb and Hyndman 2014) to conduct hierarchical load forecasting. Although these boosting techniques can be used for regression problems, they usually build the ensemble with weak base predictors such as decision trees. Their prediction accuracy may be degraded in employing relatively strong base predictors, such as autoregressive (AR) or radial basis function (RBF) models in the ensemble, because strong base predictors are prone to resulting in overfitting (Breiman, 1999).

To tackle the aforementioned overfitting problems, a novel regulated boosting technique, rBoost in short, is proposed in this work for more accurate system state forecast and material property prognosis. It is new in the following aspects: 1) a novel weight regulator is proposed in the proposed rBoost technique to effectively reduce the overfitting problem; 2) the forecast convergence of the proposed rBoost technique is analytically investigated. The effectiveness of the proposed rBoost predictor is verified by simulations. Furthermore, the new rBoost predictor is implemented for material fatigue prognosis.

The remainder of this paper is organized as follows: Section 2 presents the theoretical foundation of the proposed rBoost technique. In Section 3, the effectiveness of the proposed rBoost predictor is first

* Corresponding author.
E-mail addresses: Derek.Li@emechsys.com (D.Z. Li), wwang3@lakeheadu.ca (W. Wang), fmismail@uwaterloo.ca (F. Ismail).

http://dx.doi.org/10.1016/j.engappai.2016.09.011
Received 12 April 2016; Received in revised form 10 August 2016; Accepted 3 September 2016
0952-1976/

examined by simulations, and then it is implemented for material fatigue prognosis; finally, some concluding remarks of the study are given in Section 4.

2. The proposed rBoost technique

The motivation in rBoost technique is to incorporate more base predictors in the ensemble, each having an appropriate weight factor; correspondingly, the resulting ensemble predictor could outperform each base predictor and become a stronger predictor. The development of the rBoost technique is discussed below.

2.1. The regulated boosting in rBoost technique

Consider the training samples $z(k)$; k=1, 2, ..., N_t, where N_t is the number of samples in the training data set. For an s-step-ahead prediction, the training data set is re-arranged to have the input vector $x(i)$=[$z(i-1)$, $z(i-2)$, ..., $z(i-r)$], and the output $y(i)$=$z(i+s-1)$, where i=1, 2, ..., N; N =(N_t-r-s+1), and r is the dimension of the input vector $x(i)$.

In the ensemble of the rBoost, the base predictor h_t incorporated at step t aims to classify the training data with distribution L_t. Given the distribution L_t, the update of the distribution at step t+1 will be performed by

$$L_{t+1}(i) = \frac{L_t(i)\exp(\eta_t\beta_t|y_d(i) - p_t(i)|)}{Z_t} \tag{1}$$

where p_t are the predicted values at step t using the base predictor h_t; y_d are the desired values; β_t is the weight of the base predictor h_t and $Z_t = \sum_{i=1}^{N} L_t(i)\exp(\eta_t\beta_t|y_d(i) - p_t(i)|)$. η_t is a new weight regulator to improve error convergence and prevent the boosting from overfitting, which is given by

$$\eta_t = \frac{2 - \exp(-\zeta\Phi_t)}{\sum_{l=1}^{t}\beta_l} \tag{2}$$

where $\zeta \in (0, +\infty)$ is the factor to regulate the weight update rate. The average of root mean square error (RMSE) change rates at step t-1 and step t-2 is

$$\Phi_t = \frac{1}{2}\left(\left|\frac{\phi_t - \phi_{t-1}}{\phi_{t-1}}\right| + \left|\frac{\phi_{t-1} - \phi_{t-2}}{\phi_{t-2}}\right|\right) \tag{3}$$

where ϕ_t is the RMSE at step t, calculated by

$$\phi_t = \sqrt{\frac{\sum_{i=1}^{N}|y_d(i) - p_t(i)|^2}{N}} \tag{4}$$

If Φ_t is large, then η_t is large, in order to speed up error convergence; if Φ_t is small, then η_t is small to reduce overfitting problem. If Φ_t is constant, a larger ζ will lead to a larger η_t, the error convergence improvement becomes more significant.

By setting the initial distribution $L_1(i) = \frac{1}{N}$, i=1, 2, ..., N, similarly, the update of the distribution is carried out by

$$L_{t+1}(i) = \frac{\exp(\sum_{t=1}^{T}\eta_t\beta_t|y_d(i) - p_t(i)|)}{N\prod_{t=1}^{T}Z_t} \tag{5}$$

The weight β_t of the base predictor h_t will be

$$\beta_t = \frac{1}{2M_t}\ln\left(\frac{M_t - \lambda_t}{M_t + \lambda_t}\right) \tag{6}$$

where $\lambda_t = \sum_{i=1}^{N} L_t(i)|y_d(i) - p_t(i)|$; $M_t = \sup_i |y_d(i) - p_t(i)|$ is the maximum value of $|y_d(i) - p_t(i)|$. The detailed derivation of β_t will be discussed in the Section 2.3.

The final ensemble predictor will be derived as

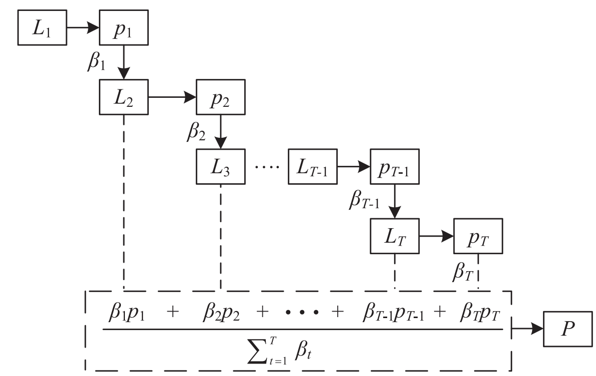

Fig. 1. Processing diagram of rBoost.

$$P = \sum_{t=1}^{T}\beta_t' p_t \tag{7}$$

where $\beta_t' = \frac{\beta_t}{\sum_{l=1}^{T}\beta_l}$ is the normalized weight of base predictors. The processing diagram of rBoost is illustrated in Fig. 1.

2.2. Derivation of the mean square error measure

Since $\beta_t \le 0$, $\eta_t < 0, \beta_t' \in [0, 1]$, $\sum_{t=1}^{T}\beta_t' = 1, \eta_t\beta_t \ge \beta_t' \ge 0$ $\sum_{i=1}^{N} L_{T+1}(i) = 1$, and the final ensemble prediction $P(i) = \sum_{t=1}^{T}\beta_t' p_t(i)$, the mean square error (MSE) of the training data can be determined by

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_d(i) - P(i))^2 \tag{8}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\left(y_d(i) - \sum_{t=1}^{T}\beta_t' p_t(i)\right)^2 \tag{9}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{t=1}^{T}\beta_t'(y_d(i) - p_t(i))\right)^2 \tag{10}$$

$$\le \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{t=1}^{T}\beta_t'|y_d(i) - p_t(i)|\right)^2 \tag{11}$$

$$< \frac{1}{N}\sum_{i=1}^{N}\exp\left(\sum_{t=1}^{T}\beta_t'|(y_d(i) - p_t(i))|\right) \tag{12}$$

$$\le \frac{1}{N}\sum_{i=1}^{N}\exp\left(\sum_{t=1}^{T}\eta_t\beta_t|y_d(i) - p_t(i)|\right) \tag{13}$$

$$= \sum_{i=1}^{N} L_{T+1}(i)\prod_{t=1}^{T}Z_t \tag{14}$$

$$= \prod_{t=1}^{T}Z_t \tag{15}$$

2.3. MSE upper bound convergence

Let $u_t = |y_d(i) - p_t(i)|$ with $u_t \in [0, M_t]$, then $\lambda_t = \sum_{i=1}^{N} L_t(i)u_t(i)$. The upper bound of Z_t can be derived as

$$\begin{aligned} Z_t &= \sum_{i=1}^{N} L_t(i)\exp(\beta_t u_t(i)) \\ &\le \sum_{i=1}^{N} L_t(i)\left(\tfrac{M_t + u_t(i)}{2M_t}\exp(\beta_t M_t) + \tfrac{M_t - u_t(i)}{2M_t}\exp(-\beta_t M_t)\right) \\ &= \tfrac{1}{2}\exp(\beta_t M_t) + \tfrac{\lambda_t}{2M_t}\exp(\beta_t M_t) + \tfrac{1}{2}\exp(-\beta_t M_t) - \tfrac{\lambda_t}{2M_t}\exp(-\beta_t M_t) \end{aligned} \tag{16}$$

Let $U = \frac{1}{2}\exp(\beta_t M_t) + \frac{\lambda_t}{2M_t}\exp(\beta_t M_t) + \frac{1}{2}\exp(-\beta_t M_t) - \frac{\lambda_t}{2M_t}\exp(-\beta_t M_t)$; the minimization of the upper bound U with respect to β_t can be achieved by setting $\frac{\partial U}{\partial \beta_t} = 0$, and after re-arranging yields

$$\beta_t = \frac{1}{2M_t}\ln\left(\frac{M_t - \lambda_t}{M_t + \lambda_t}\right) \tag{17}$$