Applied Soft Computing 22 (2014) 629–637

Contents lists available at ScienceDirect

Applied Soft Computing

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

Robust feedforward and recurrent neural network based dynamic weighted combination models for software reliability prediction

Pratik Roy[a], G.S. Mahapatra[b,*], Pooja Rani[b], S.K. Pandey[b], K.N. Dey[a]

[a] Department of Computer Science and Engineering, University of Calcutta, Kolkata 700009, India
[b] National Institute of Technology – Puducherry, Karaikal 609605, India

ARTICLE INFO

Article history:
Received 10 January 2013
Received in revised form 29 March 2014
Accepted 11 April 2014
Available online 26 April 2014

Keywords:
Software reliability growth model
Dynamic weighted combination model
Artificial neural network
Genetic algorithm
Reliability prediction

ABSTRACT

Traditional parametric software reliability growth models (SRGMs) are based on some assumptions or distributions and none such single model can produce accurate prediction results in all circumstances. Non-parametric models like the artificial neural network (ANN) based models can predict software reliability based on only fault history data without any assumptions. In this paper, initially we propose a robust feedforward neural network (FFNN) based dynamic weighted combination model (PFFNND-WCM) for software reliability prediction. Four well-known traditional SRGMs are combined based on the dynamically evaluated weights determined by the learning algorithm of the proposed FFNN. Based on this proposed FFNN architecture, we also propose a robust recurrent neural network (RNN) based dynamic weighted combination model (PRNNDWCM) to predict the software reliability more justifiably. A real-coded genetic algorithm (GA) is proposed to train the ANNs. Predictability of the proposed models are compared with the existing ANN based software reliability models through three real software failure data sets. We also compare the performances of the proposed models with the models that can be developed by combining three or two of the four SRGMs. Comparative studies demonstrate that the PFFNNDWCM and PRNNDWCM present fairly accurate fitting and predictive capability than the other existing ANN based models. Numerical and graphical explanations show that PRNNDWCM is promising for software reliability prediction since its fitting and prediction error is much less relative to the PFFNNDWCM.

1. Introduction

Computer plays an essential role in our modern civilization. Computer systems are managed by software and hence software should be reliable and fault free. Perfect measurement of software reliability has become one of the most important tasks for development of good quality software. According to ANSI definition, software reliability is defined as the probability of failure-free software operation for a specified period of time in a specified environment [1,4]. Software reliability is the most important quality metric. One can estimate the duration of software testing period by measuring software reliability. Software reliability prediction plays a vital role in today's rapidly growing complex software development process. In the last three decades, many SRGMs have been proposed in the literature to predict the relationship between software failure and time. The software reliability models can be divides into parametric and non-parametric model. In parametric models, the model parameters are estimated based on the assumptions about the behavior of the software faults, failure processes and development environments. The most well-known parametric models are the nonhomogeneous Poisson process (NHPP) based SRGMs [9–18]. However, it has been shown that none single model can produce accurate prediction results in all circumstances [19]. On the other hand, non-parametric model like the ANN based model comprise the flexibility to predict software reliability based on only fault history data without any assumptions of the parametric model. It has been exposed that ANN based non-parametric models can produce better predictive quality than the parametric models [6–8,24]. Different types of FFNNs and RNNs have been applied to predict cumulative number of detected faults where the software execution time is used as the input to the network. Several approaches have been developed to combine various existing software reliability models to produce a dynamic weighted combination model whose prediction accuracy is much better than the component models [7,20,21].

* Corresponding author. Tel.: +91 9433135327; fax: +91 4368231665.
E-mail addresses: pratik.roy43@gmail.com (P. Roy), g_s_mahapatra@yahoo.com (G.S. Mahapatra), kndey55@gmail.com (K.N. Dey).

http://dx.doi.org/10.1016/j.asoc.2014.04.012
1568-4946/

In this paper, firstly we propose a robust FFNN based dynamic weighted combination model for software reliability prediction. The PFFNNDWCM is constructed by combining the traditional SRGMs. The SRGMs are merged based on the dynamically evaluated weights determined by the training algorithm of the FFNN. In this study, we select four well-known traditional parametric SRGMs to develop the PFFNNDWCM. Based on this proposed FFNN architecture, we also propose a robust RNN based dynamic weighted combination model for more accurate prediction of software reliability. We construct the proposed FFNN and RNN by using different activation functions for hidden layer neurons of the ANNs. Cumulative software execution time is the input and the predicted cumulative number of software failures is the output of the proposed networks. We propose the RNN modeling approach to learn the temporal patterns of the failure data dynamically which has a significant impact on network prediction performance. We propose a real-coded GA based learning algorithm to train the proposed ANNs using the software failure data sets by globally optimizing the weights and parameters of the ANNs. We compare the performances of the proposed models with the existing ANN based dynamic weighted combination model [7] and ANN ensemble model [8] in the software reliability literature. The proposed models are also compared with the models that can be derived by using three or two of the selected four models. We explain the fitting and predictability of the different models through three real software failure data sets.

The rest of the paper has been organized in the following way: Section 2 introduces some related works about the application of ANNs in software reliability prediction. Section 3 presents the proposed FFNN and RNN based modeling approach for the development of the dynamic weighted combination models for software reliability prediction. In Section 4, we propose GA based learning algorithm to train the proposed FFNN and RNN. Section 5 describes the nature of the software failure data sets used in the experiments. Section 6 describes the model comparison criteria which are used to compare the prediction performance of the different models. Section 7 shows the experimental results based on the three real software failure data sets and some conclusions are drawn in Section 8.

2. Literature survey

This section describes some related works where ANNs [2] have been applied in software reliability modeling and prediction. Karunanithi et al. [6,22] first proposed ANNs to predict the software reliability by using the execution time and the cumulative number of detected faults as the input and the desired output of the network, respectively. Sitte [23] compared the predictive capability of two different software reliability prediction methods: neural networks and recalibration for parametric models. Khoshgoftaar and Szabo [25] applied ANN to predict the number of faults in programs during testing. Cai et al. [26] proposed the back-propagation algorithm based ANN for software reliability prediction and used the recent 50 inter-failure times as input to predict the next failure time. Ho et al. [27] proposed a modified RNN for modeling and prediction of software failures. Tian and Noore [28,29] proposed an evolutionary ANN modeling approach to predict the software cumulative failure time based on multiple-delayed-input single-output architecture. Su and Huang [7] proposed an ANN based dynamic weighted combination model for software reliability prediction. Hu et al. [30] applied RNNs to model fault detection and fault correction processes together. Kiran and Ravi [31] proposed a non-linear ensemble-based approach for software reliability prediction. Kapur et al. [32] applied ANN methods to build SRGMs considering faults of different complexity. Zheng [8] used the ensemble of ANNs to predict software reliability. Kapur et al. [33] presented a new dimension to build an ensemble of different ANN for complex software architectures to improve the estimation accuracy. Li et al. [20] proposed adaboosting-based approaches for combining the parametric SRGMs to significantly improve the estimating and forecasting accuracy. Wu et al. [21] proposed a dynamically-weighted software reliability combination model to improve the predictive quality. Mohanty et al. [34] proposed novel recurrent architectures for genetic programming and group method of data handling to predict software reliability.

Table 1
Selected SRGMs with mean value function.

SRGM	Mean value function
Goel–Okumoto model (G)	$a(1-e^{-bt})$
Yamada delayed s-shaped model (Y)	$a(1-(1+bt)e^{-bt})$
Inflection s-shaped model (I)	$\frac{a(1-e^{-bt})}{1+\beta e^{-bt}}$
Logistic growth curve model (L)	$\frac{a}{1+be^{-ct}}$

3. Proposed ANN based software reliability models

We know that none single SRGM can be trusted to produce accurate prediction results in all circumstances. Here, we propose FFNN and RNN-based dynamic weighted combination models which combine the traditional SRGMs to form a dynamic weighted combination of software reliability models.

We consider four well-known traditional statistical SRGMs namely, Goel–Okumoto model [9], Yamada delayed s-shaped model [16], inflection s-shaped model [14] and logistic growth curve model to develop the dynamic weighted combination models which can combine the SRGMs based on the dynamically assigned weights determined by the training of the proposed ANNs.

The mean value function of the selected four SRGMs are given in Table 1.

3.1. PFFNNDWCM

PFFNNDWCM is constructed as shown in Fig. 1 to develop the dynamic weighted combination model by combining the selected four SRGMs. The proposed FFNN has one hidden layer with single neuron in each of the input and output layers. The hidden layer consists of four neurons representing the four SRGMs to be combined.

The cumulative software execution time (t_i) is the input of the FFNN and output of the FFNN is the predicted cumulative number of software failures (y'_i), where i is the cumulative software execution time sequence index. We use $1-e^{-x}$, $1-(1+x)e^{-x}$,

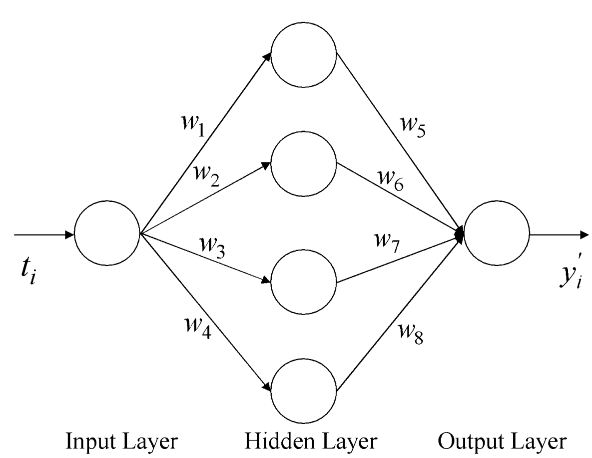

Fig. 1. PFFNNDWCM architecture.