

Software Reliability Prediction Based on Ensemble Models

Pravas Ranjan Bal, Nachiketa Jena and Durga Prasad Mohapatra

Abstract Software reliability is the determinant factor of software reliability prediction and software quality estimation during software testing period. This report offers an ensemble technique model of different artificial neural networks for forecasting of software reliability. The experimental results of the proposed model are compared with other states of the traditional models and it is noted that the proposed architectural model outperforms its competent models. The proposed architectural ensemble model has been adequately tested on three benchmark datasets and its results tested with an artificial neural network approach and a mathematical linear model. The experimental result demonstrates that the ensemble model yields better performance than other models.

Keywords Ensemble model · Feed forward neural network · Radial basis function · Statistical model and software reliability

1 Introduction

According to ANSI, software reliability is determined as the probability of software failure-free operation for a specific point of time in a conditioned environment [1, 2]. In the last four decades, most of the linear prediction models such as Software Reliability Growth Models (SRGMs) have been designed for software reliability prediction, weather forecasting, cost estimation and its time factors. Parametric neural network models are established on the linear Non-Homogeneous

P.R. Bal (✉) · Nachiketa Jena · D.P. Mohapatra
Department of Computer Science and Engineering,
National Institute of Technology, Rourkela, Rourkela 769 008, India
e-mail: pravasranjan90@gmail.com

Nachiketa Jena
e-mail: nachiketa.jena@gmail.com

D.P. Mohapatra
e-mail: durga@nitrrkl.ac.in

Poisson Process (NHPP). Lastly, it has been concluded that all parametric linear models cannot predict efficiently in all circumstances. Most of the nonparametric statistical prediction models such as artificial neural networks like Feed Forward Neural Network (FFNN) can predict different types of software reliability metrics like availability, time between failures and cumulative failures, etc. At last, it is concluded that the ensemble prediction architectural model is superior to other types of the parametric linear models and has also better prediction capability [3–5].

This work presents a non-parametric architectural model for software reliability forecasting based on ensemble techniques which can predict efficiently software reliability data than other neural network approach and linear mathematical parametric model like Duane growth model [6].

The remainder of the paper is organized as follows. In Sect. 2, some related works for forecasting of software reliability of cumulative failure data are presented. Section 3 discusses the proposed ensemble approach for software reliability forecasting. The experimental results and discussion is presented in Sect. 4. Finally, Sect. 5 concludes the paper.

2 Literature Survey

This section briefly discusses some related works based on various types of artificial neural network models for prediction of software reliability.

Karunanithi et al. [5] presented connectionist model based on feed forward and recurrent neural network. They observed that this model works well in all circumstances for different types of software reliability datasets than some analytical models.

Su et al. [4] proposed an artificial neural network modeling approach such as Dynamically Weighted Combinational Model (DWCM) for prediction of software failure history data and software reliability estimation. He compared his proposed model with some mathematical models and proved that his proposed model accurately predict than other linear mathematical models.

Site [7] used two types of software reliability prediction growth models such as FFNN and recalibration of analytical growth model for prediction. He claimed that both prediction models have better prediction capability for common datasets in all circumstances.

Tian and Noore [8] proposed an evolutionary connectionist approach for forecasting of cumulative software failure data. He used multiple delayed input and single output neural network architectural model. He observed that his proposed model performs efficiently than other prediction model.

Cai et al. [9] observed the effectiveness of an artificial neural network for software reliability prediction from software failure history data and found several new things such as the neural network modeling approach is the best approach to handling software failure data and smoothly trends than other traditional models. The training results of neural network modeling approach are more appropriate than

other linear models. He also observed that neural network modeling approach is quantitatively bad for prediction of software defect prone and qualitatively good for classifying different modules of software.

Related work reveals that most of the prediction growth models used SRGMs. However, use of ensemble technique is limited. So, this work specially focuses only on ensemble technique of three standard artificial neural network models with a goal to obtain efficient and accurate prediction.

3 Proposed Work

This section portrays the proposed model for prediction of software failure data which utilizes ensemble techniques of the artificial neural network.

3.1 Ensemble Model

The prediction model based on the Ensemble technique is depicted in Fig. 1. Ensemble model is a three layer single input and single output architectural model such as an input layer, a component layer consists of feed forward, radial basis function neural network and an output layer is an average combination of output of all component layers. The component layer consists of three components and have used three types of activation functions for three components. The cumulative software failure data are organized in pair $\{T_i, N_i'\}$, where T_i is the execution time of software failure dataset as input of the ensemble model and N_i' is the cumulative

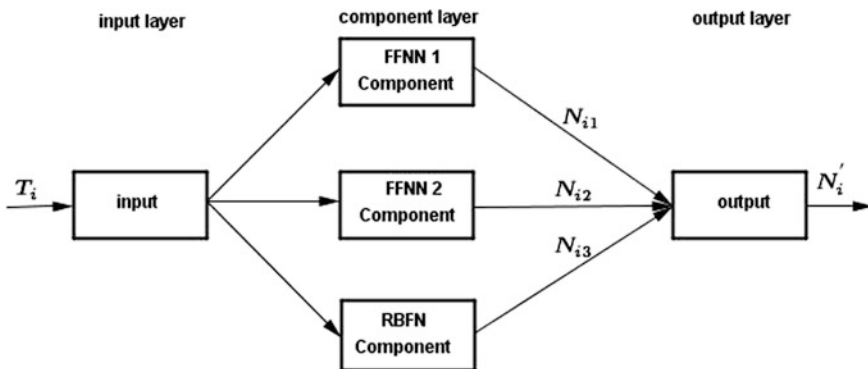


Fig. 1 The architecture of ensemble model

number software failure data as output of the ensemble model. The output of the Ensemble model is the mean of all three neural network components and is defined as follows.

$$N'_i = \frac{N_{i1} + N_{i2} + N_{i3}}{3} \tag{1}$$

3.2 FFNN Component

Two types of FFNNs have been used for our Ensemble model such as FFNN1 and FFNN2. The FFNN model is shown in Fig. 2.

The node in the FFNN model is computed as the sum of weighted sum of input data and bias value and the mathematical definition of this process is defined in Eq. (2)

$$a_i = \sum_{j=1}^n w_{ij}x_j + b_i \tag{2}$$

$$y_i = f_i(a_i)$$

where a_i is the linear combination of input data and bias value b_i and w_{ij} is the weight matrices of FFNN model.

For FFNN1 we have used the transfer function such as log sigmoid as activation function and is defined in Eq. (3)

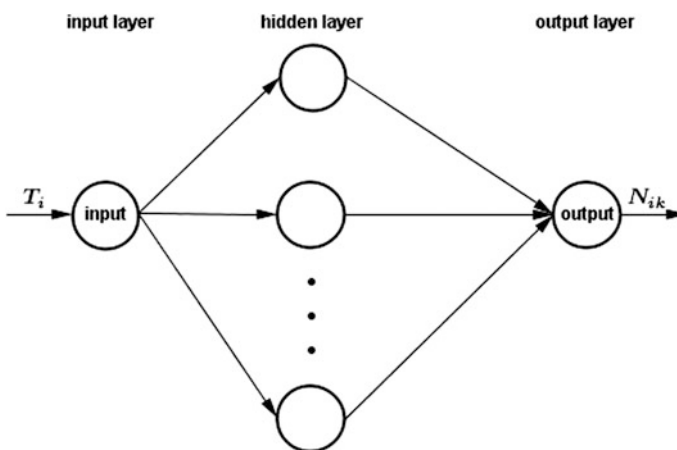


Fig. 2 FFNN model

$$f(n) = \frac{1}{1 + e^{-n}} \quad (3)$$

For FFNN2 we have used the transfer function such as tan sigmoid as activation function and is defined in Eq. (4)

$$f(n) = \frac{2}{1 + e^{-2*n}} - 1 \quad (4)$$

3.3 RBFN Component

Instead of using sigmoid function in FFNN, we have used the transfer function as radial basis function in the hidden layer of FFNN [10, 11] and defined as follows

$$f(y) = \sum_{i=1}^k w_i \phi(x - c_i) \quad (5)$$

where k is the number of neurons in the hidden layer, x is the number software failure data as input, w_i is the weight matrices of respective neuron i and c_i is the centroid vector for neuron i .

The radial basis function $\phi(x - c_i)$ as given by

$$\phi(x - c_i) = \sqrt{\sum_{i=1}^k (x - c_i)^2} \quad (6)$$

3.4 Performance Measures

Two types of meaningful performance measures have been used to compare the reliability prediction error of ensemble model and its competent models. Here, the proposed ensemble model is trained with some part of software failure data and the rest of software failure data is used for testing purpose. For performance measurement we have used two types of errors called Relative Error (RE) and Average Error (AE) are defined as

$$RE = \frac{(\hat{y}_i - y_i)}{y_i} * 100 \quad (7)$$

$$AE = \frac{1}{n} \sum_{i=1}^n RE_i \tag{8}$$

where, n is the total number of data samples, \hat{y}_i is the predicted value and y_i is the actual value.

4 Experimental Results and Comparison

In our experiment, this model is trained and tested with three benchmark datasets DS1, DS2 and DS3 [1] for software reliability forecasting. The dataset DS1 consists of 21,700 numbers of assembly instructions and 136 numbers of failures and collected from real-time command and control application. The dataset DS2 consists of 10,000 assembly instructions and 118 numbers of failures and collected from flight dynamic application. The dataset DS3 consists of 22,500 assembly instructions and 180 numbers of failures and collected from flight dynamic application. All these datasets have been normalized in the range between [0, 1] by min max formula.

The ensemble model is trained with 60, 60 and 55 % for DS1, DS2 and DS3, respectively, in our experiments. The remaining software data for each software failure history dataset is used for testing. We have used different training ratio of different datasets for better prediction results.

We compare our proposed ensemble model with one artificial neural network such as FFNN and another linear mathematical parametric model [6] called Duane growth model: The Duane growth model is given by $\beta(t) = bt^a$ $b > 0, a > 0$, where a is the is shape of the growth curve and b is the parameter size of the curve.

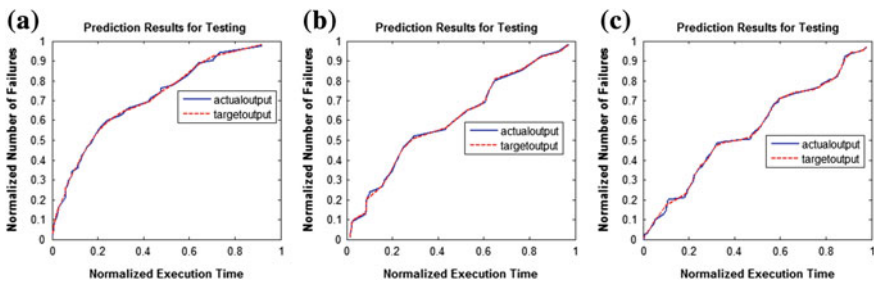


Fig. 3 Prediction results of ensemble model for a DS1, b DS2 and c DS3

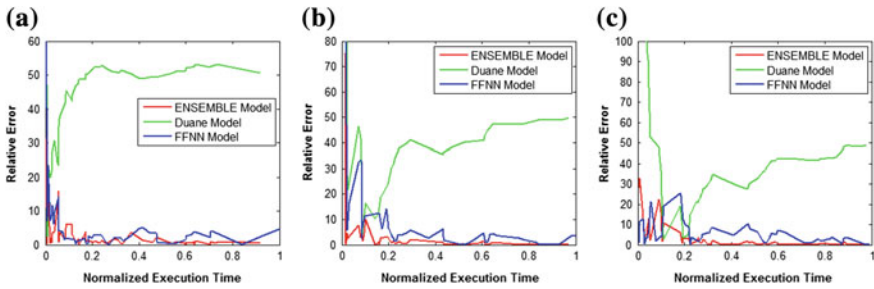


Fig. 4 Relative error of different models for **a** DS1, **b** DS2 and **c** DS3

Table 1 AEs for different models

Dataset	Average error		
	Ensemble model	FFNN	Duane model
DS1	1.1519	2.2156	47.2190
DS2	1.2540	3.1169	30.4239
DS3	1.1236	4.2752	21.1891

4.1 Performance Comparison

For ensemble model, we have chosen three components such as FFNN1, FFNN2, RBFN and each component consists of 5 neurons in the hidden layer. For FFNN model, we have taken five neurons in the hidden layer. The prediction results of Ensemble model for DS1, DS2 and DS3 are depicted in Fig. 3. The REs of different model on three datasets DS1, DS2, and DS3 are depicted in Fig. 4. The AEs on three datasets are shown in Table 1. It is concluded that ensemble modeling approach shows better performance than FFNN model and another linear mathematical modeling approach. For all datasets linear parametric growth model called Duane model shows worse performance than others. It is observed from the above results and discussion the Ensemble model is better than alternate traditional growth models.

5 Conclusion

In this paper, a non-parametric prediction approach has been proposed such as ensemble technique of three different artificial neural networks for software reliability prediction. The proposed ensemble model shows better prediction than other artificial neural network model and another traditional parametric mathematical model. It is observed that from the experimental results and discussions the proposed assembling approach proves the best model and shows lower prediction error than other competent model.

References

1. Lyu, M. R.: Handbook of software reliability engineering. New York: McGraw-Hill (1996).
2. Musa, J. D.: Software reliability engineering. More reliable software, faster development and testing. New York: McGraw-Hill (2004).
3. Zheng, J.: Predicting software reliability with neural network ensembles. Expert System with applications 36.2. (2009) 2116–2122.
4. Su, Y. S., Huang, C. Y.: Neural-network-based approaches for software reliability estimation using dynamic weighted combinational models. Journal of Systems and Software, 80, (2007) 606–615.
5. Karunanithi, N., Whitley, D., Malaiya, Y. K.: Using neural networks in reliability prediction. IEEE Software, 9 (1992) 53–59.
6. Malaiya, Y. K., Li, M. N., Bieman, J. M., Karcich, R.: Software reliability growth with test coverage. IEEE Transactions on Reliability, 51 (2002) 420–426.
7. Sitte, R.: Comparison of software-reliability-growth predictions: neural networks vs. parametric recalibration. IEEE Transactions on Reliability, 48(3) (1999) 285–291.
8. Tian, L., Noore, A.: On-line prediction of software reliability using an evolutionary connectionist model. Journal of Systems and Software, 77 (2005) 173–180.
9. Cai, K. Y., Cai, L., Wang, W. D., Yu, Z. Y., Zhang, D.: On the neural network approach in software reliability modeling. Journal of Systems and Software, 58 (2001) 47–62.
10. Orr, Mark, J. L.: Introduction to radial basis function networks (1996).
11. Leonard, J. A., Kramer, M. A., Ungar, L. H.: Using radial basis functions to approximate a function and its error bounds. IEEE transactions on neural networks/a publication of the IEEE Neural Networks Council, 3(4) (1991) 624–627.