Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 92 (2016) 418 – 424

Procedia Computer Science

2nd International Conference on Intelligent Computing, Communication & Convergence

(ICCC-2016)

Srikanta Patnaik, Editor in Chief

Conference Organized by Interscience Institute of Management and Technology

Bhubaneswar, Odisha, India

Software Quality Prediction Model with the Aid of Advanced Neural Network with HCS

[1]Kavita Sheoran*, [2]Dr. Pradeep Tomar, [3]Dr. Rajesh Mishra

[1]Reader, Department of Computer Science ,MSIT,Delhi
kavitasheoran0780@gmail.com
[2]Assistant Professor in Department *of Computer Science and Engineering,*
School of Information and Communication Technology,
Gautam Bhudda University, Greater Noida
[3]Assistant Professor , School of ICT ,
Gautam Buddha University, Greater Noida,,Uttar Pardesh

Abstract

Software quality is regarded as the highly important factors for assessing the global competitive position of any software product. To assure quality, and to assess the reliability of software products, many software quality prediction models have been proposed in the past decades. In this proposed method we have utilized a hybrid method for quality prediction. The prediction is done with the help of the Advanced Neural network which is incorporated with Hybrid Cuckoo search (HCS) optimization algorithm for better prediction accuracy. The application software is first subjected to test case generation and once the test cases are generated they are applied to advanced neural network for the prediction of quality. The neural network is improved by utilizing HCS which optimizes the weight factor for improving the prediction. The quality metrics like maintainability and reliability are estimated for predicting the software quality and the results are compared with other existing techniques to verify the effectiveness of our proposed method.

Peer-review under responsibility of the Organizing Committee of ICCC 2016

Keywords: Software quality prediction, Cuckoo search algorithm, Neural Network, Optimization, Test cases.

* Corresponding author. Tel.: 91-9818171914
E-mail address:kavitasheoran0780@gmail.com

Peer-review under responsibility of the Organizing Committee of ICCC 2016
doi:10.1016/j.procs.2016.07.365

1. Introduction

The software quality models play a very significant part as highly fruitful devices dedicated for the realization of the vital targets of a software quality assurance scheme [1] [2].The software measures constitute the measurements of diverse features of software like the dimension, intricacy and the association between its segments [7].Classically, the software modules are classified by a categorization model into two diverse risk-based groups, like the fault-prone (fp), and not fault-prone (nfp) modules. [3]. The quality-based class membership of the modules in the training data set is habitually allocated in accordance with a pre-set threshold value of a quality factor like the number of errors, or the number of lines of code churns [4]. In the process, the software developer's scant resources may be dedicated towards the identification and rectification of errors arising only in those software modules which are highly fault-prone [5]. The hunt for an optimal solution becomes further complex in the case of modeling with multiple software project data sets [6]

2. Related work

Bardsiri *et al.* [5] brilliantly brought to limelight an innovative technique devoted for the augmentation in the accuracy of development effort evaluation dependent on the integration of three diverse approaches such as the fuzzy clustering, ABE and the ANN methods. The quality of training in the ANN and the stability of the historical data in the ABE were enhanced by means of the novel framework. Monden *et al.* [6] made history by majestically launching an innovative simulation model of software testing so as to evaluate the cost-consciousness of test effort allocation technique dependent on the error forecast outcomes. Rahimi and Zargham [7] remarkably launched a vulnerability scrying, a paradigm for the vulnerability discovery forecast in accordance with the code properties. Thereafter, they brought in a stochastic model which employed code features as its constraints to forecast the vulnerability discovery.

3.1 Process involved in the proposed software quality prediction technique

The quality prediction is basically performed based on the estimation of various quality attributes in the software. The proposed method utilizes an advanced neural network classifier for the classification process which is optimized using the hybrid Cuckoo search optimization algorithm for better classification process. The fig 1 given below shows the entire process flow involved in the proposed method.

3.2. Test case generation

Test cases are employed to test all feasible combinations in the application and as well it offers the user to simply replicate the steps that were assumed to expose a defect that as identified during test. Test cases can be charted directly and obtained from use cases. Moreover, when the test cases are produced early, Software Engineers can frequently discover ambiguities and inconsistencies in the requirements specification and design documents. The generated test cases will be fed to the advanced neural network for classification based on which the software quality will be predicted.

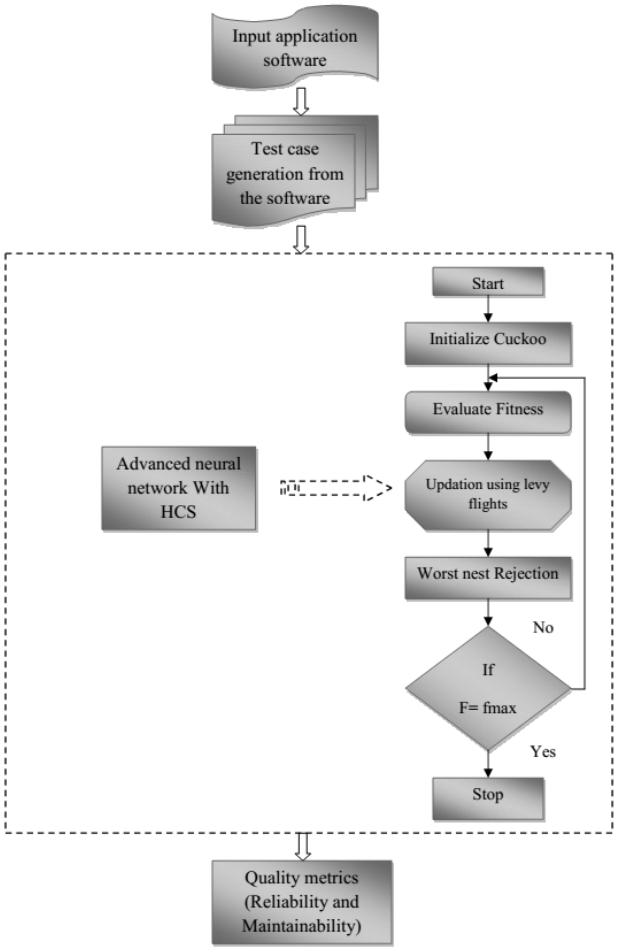

Fig 1: Proposed quality prediction technique

3.3 Classification using Advanced Neural network with HCS

The Advanced Neural Network is utilized to ascertain the software quality prediction process and it is trained by employing the testcases that are generated from the input application software. The advanced neural network consists of three input units, n hidden units and one output unit. The network is trained under a large set of testcases in order to enable them to effectively predict the quality measure in the testing phase

3.3.1 Weight Optimization Using Modified cuckoo search algorithm

The Cuckoo search algorithm represents a meta-heuristic algorithm which owes its origin to the breeding conduct of the cuckoos and it is easy of implementation The modus operandi of the clustering procedure is shown as follows:

Step 1: *Initialization Phase:* The population p_i , where i=1, 2, n of host nest is initiated arbitrarily.

Step 2: *Generating New Cuckoo Phase:* With the help of the levy flights a cuckoo is selected randomly which generates novel solutions. Subsequently, the engendered cuckoo is evaluated by employing the objective function for ascertaining the excellence of the solutions.

Step 3: *Fitness Evaluation Phase:* The fitness function is evaluated in accordance with Equations 1 and 2 shown hereunder, followed by the selection of the best one.

$$R_{\max} = \frac{R_u}{R_v} \tag{1}$$

$$F = \max imum\ popularity = R_{\max} \tag{2}$$

Where, R_u - signifies the selected population and R_v - represents the total population

Step 4: *Updation Phase:* The levy flights employed for the general cuckoo search algorithm is expressed by the Equation 14 shown below:

$$p_i^{*} = p_i^{(t+1)} = p_i^{(t)} + \alpha \oplus Levy(m) \tag{3}$$

By suitably adapting Equation 3, levy flight equation using the gauss distribution is exhibited in Equation 4 here under:

$$p_i^{*} = p_i^{(t+1)} = p_i^{(t)} + \alpha \oplus \gamma_d \tag{4}$$

Where, $\gamma_d = \gamma_0 \exp(-\rho C)$ (5) and γ_0, ρ - represents the constants, C – Symbolizes the current generation

Step 5: *Reject Worst Nest Phase :* In this section, the worst nests are ignored, in accordance with their possibility values and novel ones are constructed.

Step 6: *Stopping Criterion Phase:* Till the achievement of the maximum iteration, the procedure is continued.

3.4 Software Maintainability and Reliability

Maintainability of the software is the manner by which software can be adapted and it is regarded to be the major software quality feature. Using the expression specified beneath the abstractness for the software is computed, $A = N_A / N_C$ Where, N_A = total number of abstract class in the application, N_C = total number of classes in the application , Likewise the instability is computed by means of the beneath expression, $Instability(I_n) = \frac{c_e}{(c_e + c_t)}$ Where, c_e =Efferent Coupling, c_t = Total Coupling.

The reliability of the software can be measure during the expression given as $r_s(d_t) = f(d_t)/e_{d_t}$ Where, $f(d_t)$ = Failure rate at time (d_t) e_{d_t} = execution time.

4. Results and discussion

The proposed quality prediction technique is performed using the advanced neural network with optimization algorithm involving Hybrid Cuckoo Search (HCS).The implementation done in the JAVA platform. The table 1 given below shows the fitness value of our proposed method with improved particle swarm optimization method and evolutionary programming using different iterations. The fig 2 given below shows the comparison of the fitness value. The graph shows that our proposed method has delivered better fitness value which aids in improving the quality of the software. For various time intervals the corresponding reliability and the cost values are estimated.

Table.1 Fitness value for different iteration

No of Iterations	Fitness Value		
	IPSO	Evolutionary programming	Proposed method
5	12.658	15.2	17.841
10	10.354	14.6	16.245
15	10.354	12.8	15.541
20	10.123	11.32	14.343
25	9.654	10.5	11.641

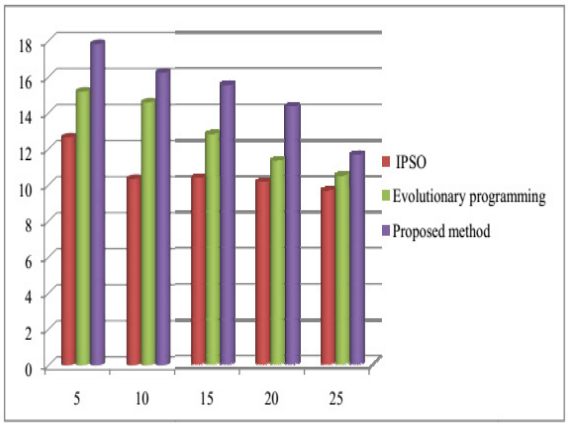

Fig 2: Comparison of the fitness value for the existing work using IPSO, Evolutionary algorithm and our proposed method

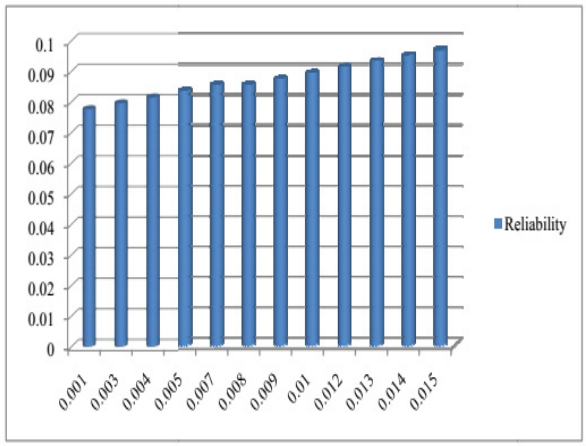

Fig.3 Reliability value of proposed method

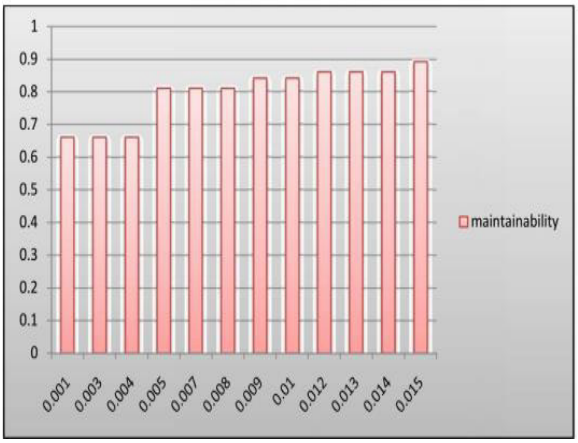

Fig.4 Maintainability value of proposed method

The fig 3 and fig 4 given above shows the graphical representation of the reliability and maintainability value for our proposed method with respect to the time. The comparative table for reliability for the proposed and existing methods where IPSO and EA are used is given in the table 2 below,

Table.2 Comparison of software quality measures for our proposed and existing method.

Methods	Reliability
IPSO	0.0294
Evolutionary Algorithm	0.03242
Proposed Method	0.08726

5. Conclusion

In this paper, we have proposed a system to predict better software quality using Advanced neural network classifier technique.. The testcases are applied to the classifier which is incorporated with hybrid cuckoo search algorithm inorder to make the classification process more accurate. Next, the quality of the software is measured with the aid of reliability and maintainability. From the comparative analysis it is clear that our proposed method achieved better quality compared to other existing methods.

Reference

1. Naeem Seliya and Taghi M. Khoshgoftaar, "Software Quality Analysis of Unlabeled Program Modules With Semisupervised Clustering",IEEE Transactions On Systems, Man, And Cybernetics—Part A: Systems And Humans, Vol. 37, No. 2, 2007.
2. Taghi M. Khoshgoftaar and Yi Liu, "A Multi-Objective Software Quality Classification Model Using Genetic Programming", IEEE Transactions On Reliability, Vol. 56, No. 2,2007
3. Taghi M. Khoshgoftaar and Kehan Gao, "Count Models for Software Quality Estimation",IEEE Transactions On Reliability, Vol. 56, No. 2,2007.
4. Z.A. Rana, M.M. Awais and S. Shamail, "Nomenclature unification of software product measures",IET Softw., Vol. 5,No. 1, pp. 83–102,2011.
5. V. Khatibi Bardsiri, D.N.A. Jawawi, S.Z.M. Hashim and E. Khatibi, "Increasing the accuracy of software development effort estimation using projects clustering",IET Softw., Vol. 6, No.6, pp. 461–473, 2012.
6. Akito Monden, Takuma Hayashi, Shoji Shinoda, Kumiko Shirai,Junichi Yoshida, Mike Barker and Kenichi Matsumoto, "Assessing the Cost Effectiveness of Fault Prediction in Acceptance Testing", IEEE Transactions On Software Engineering, Vol.39,No.10,2013.
7. Sanaz Rahimi and Mehdi Zargham, "Vulnerability Scrying Method for Software Vulnerability Discovery Prediction Without a Vulnerability Database",IEEE Transactions On Reliability, Vol. 62, No. 2,2013.

Author Biographies

Kavita Sheoran obtained his Bachelor's degree in Computer Science from University of Dr.B.R.Ambedkar University Then she obtained his Master's degree in Information Technology and pursuing PhD in Computer Science majoring in Software Engineering from Gautam Buddha University, Greater Noida. Currently, she is a Reader in Computer Science Department, MSIT,Delhi. Her current research interests are Software Engineering and Soft computing.

Dr. Pradeep Tomar is currently an Assistant Professor in the Department of Computer Science and Engineering at School of Information and Communication Technology, Gautam Buddha University, Greater Noida Distt. Gautam Budh Nagar, Uttar Pradesh, INDIA . He has published 3 papers in national journals, 5 papers in international journals, 16 papers in national conference and 6 papers in international conference.

Dr. Rajesh Mishra is currently an Assistant Professor in School of ICT , Gautam Buddha University, Greater Noida, Uttar Pradesh, INDIA . He has 5 years of teaching experience in an engineering college under Rajeev Gandhi Technical University Bhopal (MP) and VIT University Vellore (TN). He has published 4 papers in international journals, 16 papers in national conference and 10 papers in international conference.