International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 6, Issue 1, No 4, 2018. ISSN: 2348-6600
Reference ID: IJCS-342 PAGE NO: 2286-2292

SOFTWARE DEFECT DETECTION USING METAHEURISTIC GENETIC NEURAL NETWORK

R.Deepa[1], S.Gnanapriya[2],

[1]*Research Scholar,*
Department of Computer Science,
Kovai Kalaimagal College of Arts & Science,
Coimbatore-641 109, India.
rbdeepa1995@gmail.com

[2]*Assistant Professor,*
Department of Computer Science
Kovai Kalaimagal College of Arts & Science,
Coimbatore-641 109, India.
gnanapriya_2006@yahoo.co.in

Abstract- The performance of the Software defect prediction is significantly inclined by the characteristics of the software metrics which raises the efficacy of the software substantially. During the development and maintenance phase, it is very expensive to detect and rectify software module defects. This research work aims at developing an intelligent software prediction model to predict the defect in software system in an efficient manner. The dataset used in this research work is collected from NASA repository. Since the dataset is voluminous it is handled by reducing its feature subset with the use of greedy stepwise search algorithm. This algorithm selects the most prominent features which are influence the dependent variable to predict the defect and defect free modules. After feature reduction the prediction process is done by evolutionary artificial neural network. The standard artificial neural network is enhanced by applying genetic algorithm to improvise the performance of the learning phase.

The genetic algorithm fine tunes the parameters applied in hidden nodes of the hidden layer. By assigning the weights and bias with the knowledge acquired from genetic algorithm the performance of the artificial neural network in the software defect prediction produces more accurate results. This is proved by performing simulation on the proposed evolutionary artificial neural network (MGNN) using MATLAB simulator. From the result it is proved that the performance of the MGNN is better while comparing with Standard Artificial Neural Network (ANN), Probabilistic Neural Network (PNN) and Group Method of Data Handling (GMDH)

Index Terms— Genetic Neural Network, Software Defect, ANN, PNN, GMDH. *(key words)*

I. INTRODUCTION

In this modern era the growth and demanding of software applications reach great heights. As the usage of software applications increased there the need for software quality becomes a most significant issue in software engineering field. The quality of software highly depend on the software testing outcomes and this empowers the need for adapting significant testing strategy because nearly above 50% of the cost is spend for testing phase in software development Life cycle (SDLC). Testing and finding errors or defects or faults at earlier stages of SDLC will greatly reduce the computation cost and enhances the efficiency of software quality to the best level. Thus this research work concentrates on developing optimized test case generation for testing the performance of the software and determining faults or defects in module at earlier

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 6, Issue 1, No 4, 2018. ISSN: 2348-6600
Reference ID: IJCS-342 PAGE NO: 2286-2292

stage of software testing life cycle by performing efficient handling of software defect prediction process.

There are numerous data mining methods proposed by many researchers for software defect analysis, among them only a few of them are able to produce successful models to overcome the above issues. Though Classification models which have the capability to predict software faultiness is very specific and they fail to give any clues regarding the actual fault occurred in the software. Many existing works failed to select proper features to be involved for prediction process especially in the case of uncertainty among feature selection. The rule based algorithms are also handled in software defect prediction but optimal rule selection and usage is not considered by the existing approaches. All the aforementioned problems motivate to develop a three stage evidential bionic approach for optimal software defect prediction to increase the accuracy and reduce the false classification or prediction. The objective of this research work is to design and develop an evidential bio inspirational approach for optimizing the performance of the software defect prediction by overwhelming the issues in case of uncertainty in determination of defect-prone modules in software which is under consideration. The objectives are confined to the following:

- To develop a Metaheuristic based feature selection to handle the voluminous problem in software defect prediction dataset.
- To handle the uncertainty in determining the similar patterns among the dataset using Artificial Neural Network.
- Determining potential rules from the inferred knowledge using the evolutionary approach by assigning optimal weights.

II. LITERATURE REVIEW

Ahmet Okutan, et.al.(2012)[1], developed innovative method using bayesian network which exposes the relationship among software metrics. Nine different data sets from open repository has been used and it shows that RFC, LOC, and LOCQ father qualities of scooter has been determined effectively. Also another two more metrics used in this work are number of developers and LOCQ for source code quality. Finally the marginal defect probability of the selected software is measured with the effective metrics and finds the relationship among them. These kinds of defect prediction models are executed to reduce the magnitude of defects.

Supreet Kaur, et.al. (2012) [2], performed the analysis of density based spatial clustering along with the presence of noise. It was used for forecasting the errors in OOSS and C,++ language based software platforms. The feature selection process is included in forecasting the software defect.

Xiao-dong Mu et. al.,(2012)[3], in their work improved software defect prediction accuracy using evolutionary approach based on competitive organization using different machine learning algorithms. Firstly the competition mechanism is introduced to evolutionary algorithm in four different levels they are evolutionary operators, reduced operators, allied operators and distributed operators which was developed by the evolution of population.
Kamaljit Kaur (2012)[19] Prediction of fault proneness software models in the fall tips for the models are unavailable is a very big challenge in the field of software industry. Author attempted to predict the false promise of their model and the labels or not present. They used in genetic algorithm for software defect prediction approach.

Karpagavadivu et.al (2012) [4] projected novel seed methods which defends the amount of defects in the software module. Innovative method of kernel has been related and revealed that it attained companionable consequences on rbf and linear Kernels. Additionally, the scheme software defect prediction methods also compared with the surviving techniques of defect discovery methods in the works like linear regression and IBK. It was observed that the prior to the test case for maintenance phase the software developers can use this proposed method to easily predict the most defective models in the system and focusing on am primarily rather than texting each and every module in the software system. This can decrease the testing effect and the project cost automatically.

Ajeet and Neeraj [5] they proposed software fault prediction using ID3 and decision tree.

Ahmetand Olcay (2013) [6] in their work related many machine learning approaches to predict software defect based on object oriented metrics. This approach uses artificial neural network for prediction process. The Ann based prediction model performance was best in achieving the software defect prediction by utilizing the

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 6, Issue 1, No 4, 2018. ISSN: 2348-6600
Reference ID: IJCS-342 PAGE NO: 2286-2292

Object Oriented metrics. It is a better competitor of the existing approaches which uses less resources and limited time for prediction process.

Agasta Adline and Ramachandran. M (2014)[7] developed a software defect prediction model with some algorithms and analyzed their usage of software quality assurance. The objectives of fault prediction using mining techniques are clearly discussed in this work. By consuming this generous of method the cost efficiency and the time difficulty significantly condensed. By consuming this planned technique efficiently, in a restricted fashion resources are utilized and the global fault rates of entire prevailing methods are associated and benefit of the projected technique was examined in part

Shanthini and Chandrasekaran [8] developed a machine learning approach using support vector machine for software defect prediction in object oriented software. They used two different metrics namely method level and class level for one type of data set. Method level are applicable for object oriented and procedural programs. Class metrics suits for only object oriented software. Four different classifiers are used namely random forest, svm, naïve bayes and k-star.

Manu Banga, (2013) [9], in their approach new innovative computational intelligence was developed with the help of genetic programming and method of data handling. It is proved that this model is best among all other techniques based on software cost estimation in software defect prediction. For software defect prediction they developed an artificial neural network in order to generalize the concept of software defect prediction model. Further the support vector machine approach was used along with the Revolutionary and learning algorithms. The proposed method was tested with 11 different machine learning models on NASA datasets. Finally they conclude that its proposed model produces better accuracy and precision then other existing models used in their work.

Mohamad and Vahid (2014)[10] adopts determination of reusable oops models using artificial neural network technique. The software metrics are used for analyzing the structural phenomena of the software defect prediction. The training data set uses MSE, RMSE has been evaluation metrics and it was found that the present model will improve the accuracy of the software defect prediction.

III. METHODOLOGY

A. Feature Selection Method-Greedy Stepwise Search

In Greedy Stepwise Search approach it uses a heuristic decision making to perform local optima at each iteration with the confidence of determining global optima. After normalizing the software defect prediction dataset the significant features which greatly influence the class variable is discovered using greedy stepwise search algorithm. It reaches the goal after the considerable amount of time while it produces an optimal solution.

The important process in Greedy stepwise search is as follows:

- ❖ It generates candidate set for determining potential attributes.
- ❖ It applies selection function to discover best candidates.
- ❖ By applying feasibility function the most significant candidates are extracted.
- ❖ Using solution function it finds the complete solution.
- ❖

erformance Comparison on PC1 Dataset based on Feature subset generated by Rank Search Algorithm, Greedy Stepwise Search and Best First Search

TABLE I

Feature Selection Methods	Features Selected Id	No. of Features
Greedy Stepwise Search	2,9,14,15,16,18	6
Rank Search	1,2,4,5,6,9,13,14,15,16,17,18,21	13
Best First Search	1,2,3,4,5,6,7,8	8

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 6, Issue 1, No 4, 2018. ISSN: 2348-6600
Reference ID: IJCS-342 PAGE NO: 2286-2292

B. Enhanced Genetic Neural Network

After the feature subset is generated using greedy-stepwise algorithm next the training phase of the genetic-Neural network is done for prediction of software defects in modules. In this research work genetic algorithm is used for altering the weights during the learning phase of Neural Network. The number of neurons used for input is 6 input neurons in input layer with 4 hidden neurons in the hidden layer and one output layer

The Methodology of Enhanced Genetic Algorithm based Software defect Prediction collects the dataset from the PROMISE repository of PC dataset which consist of c1109 instances with 22 attributes. The selected raw dataset is preprocessed using normalization approach in which the dataset are converted to the values of the same range. The prediction of defect or no defect for the given instance is determined using the genetic algorithm in this work its performance is enhanced by adapting a new operator known as insertion operator. The genetic algorithm starts with selection process which selects the set of instances of promising result to act as the population set. From it the instances are performed with mutation and crossover to determine the different combination of the values of the given instances and the produced result are justified with the fitness value in each iteration.

The instances with highest fitness value are sustained and remaining of them are removed and new set of instances are inserted with the process of insertion operator to overcome the problem of global optimization and thus it predicts the instances falls under the defect or no defect. The below Figure 1 shows the overall Framework of Metaheuristic Genetic Neural Network.

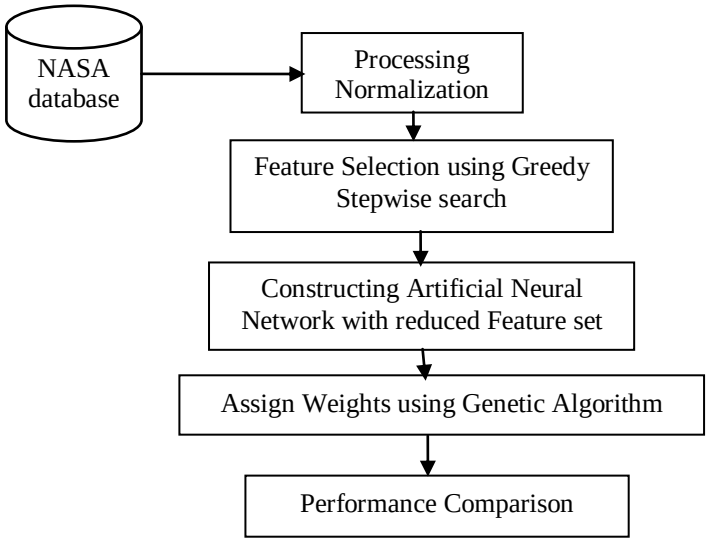

Fig. 1 Overall Framework of Genetic Neural Network

IV. EXPERIMENTAL RESULTS AND DISCUSSION

The below table II and Figure 2, 3 shows the performance comparison of the proposed method enhanced genetic algorithm with the conventional genetic algorithm and ANN. It is observed from the result that the correctly classified instances is high with 95.10 using enhanced genetic algorithm based approach in software defect prediction while the conventional genetic algorithm produces 88.9 and the worst case is performed by ANN with the value of 75.16. The highest error rate is produced by ANN because of not handling

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 6, Issue 1, No 4, 2018. ISSN: 2348-6600
Reference ID: IJCS-342 PAGE NO: 2286-2292

the correct classification of instances in software defect prediction. The proposed method with its improvement in the global optimization it produces best result in software defect prediction with high true positive rate, precision, recall and f-measure.

Performance comparison of Software Defect Prediction techniques for PC1 dataset

TABLE II

Metrics	GNN	PNN	GMDH	ANN
Correctly classified	93.0568 %	88.7286 %	89.7205 %	78.5392 %
Incorrectly classified	6.9432 %	11.2714 %	10.2795 %	21.4608 %
Mean absolute error	0.1242	0.1139	0.1039	0.2158
Root mean squared error	0.2491	0.3327	0.314	0.4262
Relative absolute error	95.573 %	87.6012 %	79.9811 %	166.0641 %
Root relative squared error	98.014 %	130.8849 %	123.5117 %	167.6689 %

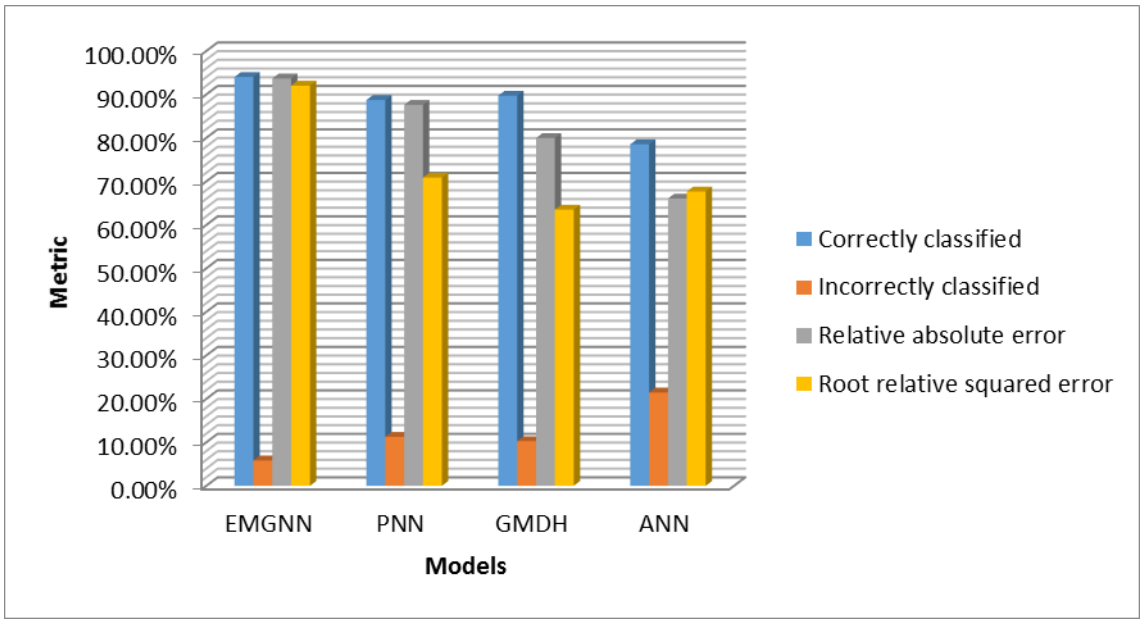

Fig. 2 Comparison of Models based on Accuracy

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 6, Issue 1, No 4, 2018. ISSN: 2348-6600
Reference ID: IJCS-342 PAGE NO: 2286-2292

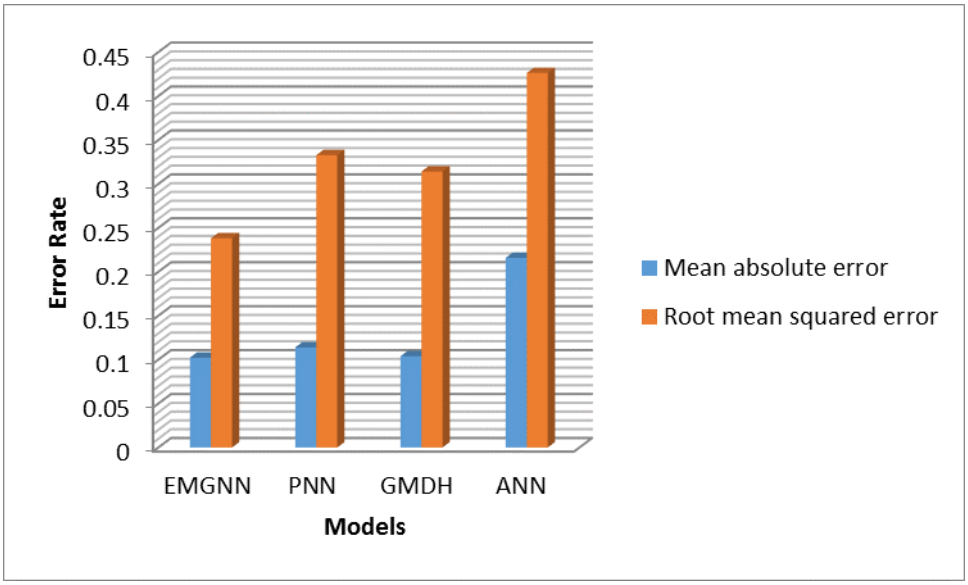

Fig. 3 Comparison of Models based on Error Rates

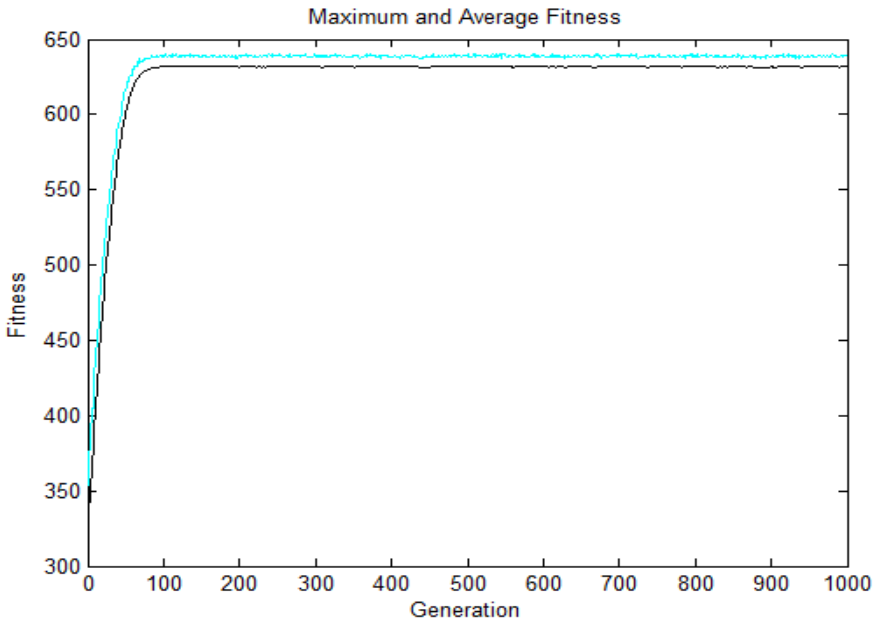

Fig. 4 Generation versus fitness value of the instances

In this figure 4 the maximum fitness value obtained during each generation is depicted the instances which hold the highest fitness value alone is selected for the next iteration and remaining of them are

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 6, Issue 1, No 4, 2018. ISSN: 2348-6600
Reference ID: IJCS-342 PAGE NO: 2286-2292

eliminated from the process and a insertion operator is used for selecting new set of population to mingle with the existing population for overcoming the problem of global optimization and thus its performance is better compared to the existing conventional genetic algorithm.

V. Conclusion

A vital role in improvising the quality of the software is fulfilled by software defect prediction. The portability of the software can assist in reducing the time taken and the cost of the product. Developing software defect detection model for software projects is helpful for reducing the effort in locating defects. In General Software defect prediction models are built based supervised learning and unsupervised learning. This work proposed the enhanced genetic algorithm based software prediction model by overcoming the problem of global search in conventional genetic algorithm by introducing the insertion operator. The performance comparison is done with Artificial Neural Network and optimizing its training process using genetic algorithm and the result shows the optimal performance of the classification of defect and non-defect instances of the software modules. This research work utilized the evolutionary based algorithm for predicting the software defect. The dataset used in this research work is collected from NASA PROMISE repository. The PC1 software defect prediction dataset is used for this proposed work performance analysis. The conventional traditional genetic algorithm is enhanced by introducing the insertion operator in the process for fitness evaluation and finding the optimal instances in the whole dataset.

In future, focus can be shifted to other techniques of data mining for defect prediction. The datasets used could also be the other Open Source datasets that are available in the Promise Repository.

Acknowledgment

I am greatly indebted to my parents and department faculties for their great encouragement and co-operation in all aspects to develop this paper. I wish to thank everyone who helped us directly or indirectly for the successful completion of this paper.

References

[1] Ahmet Okutan, Olcay Taner Yıldız,(2012) "*Software defect prediction using Bayesian networks*", Empirical Software Eng (2014) 19:154–181 © Springer Science+Business Media, LLC.

[2] Mrinal Singh Rawat, Sanjay Kumar Dubey,(2012) "*Software Defect Prediction Models for Quality Improvement: A Literature Study*", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 2, pp 288-296.

[3] Supreet Kaur, and Dinesh Kumar, "*Software Fault Prediction in Object Oriented Software Systems Using Density Based Clustering Approach*", International Journal of Research in Engineering and Technology (IJRET) Vol. 1 No.2 March,2012 ISSN: 2277-4378

[4] Mrs.Agasta Adline, Ramachandran. M(2014), "*Predicting the Software Fault Using the Method of Genetic Algorithm*", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 3, Special Issue 2,, pp 390-398.

[5] Ajeet Kumar Pandey, Neeraj Kumar Goyal," *Predicting Fault-prone Software Module Using Data Mining Technique and Fuzzy Logic*", Special Issue of IJCCT Vol. 2 Issue 2, 3, 4; 2010 for International Conference [ICCT-2010], 3rd-5th December 2010

[6] Karpagavadivu.K, et.al. (2012), "*A Survey of Different Software Fault Prediction Using Data Mining Techniques Methods*", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 1, Issue 8, pp 1-3.

[7] Kamaljit Kaur (2012), "*Analysis of resilient back-propogation for improving software process control*" International Journal of Information Technology and Knowledge Management July-December 2012, Volume 5, No. 2, pp. 377-379.

[8] Akalya devi.C, Kannammal. K.E and Surendiran.B*," A Hybrid Feature Selection Model for Software Fault Prediction*", International Journal on Computational Sciences & Applications (IJCSA) Vo2, No.2, April 2012

[9] Jie Xu, [2]Danny Ho and [1]Luiz Fernando Capretz, "*An Empirical Study On The Procedure Drive Software Quality Estimation Models*", International journal of computer science & information Technology (IJCSIT) Vol.2, No.4, (2010).

[10] Manu Banga, "*Computational Hybrids Towards Software Defect Predictions*", International Journal of Scientific Engineering and Technology Volume 2 Issue 5, pp : 311-316, (2013).