Usage Patterns and Implementation of Random Forest Methods for Software Risk and Bugs Predictions

Alankrita Aggarwal, Kanwalvir Singh Dhindsa, P. K. Suri

Abstract: The software bugs predictions whereby the datasets of different types of bugs are evaluated for further predictions. In this research manuscript, the pragmatic evaluation of random forest approach is done and compared with results with traditional artificial neural networks (ANN) so that the results can be compared. From the outcome, the extracts from random forest are better on the accuracy level with the test datasets used in a specific format. The process of Random Forest (RF) Approach is adopted in this work that gives the effectual outcomes in most of the cases as compared to ANN and thereby the usage patterns of RF are performance aware. The paradigm of RF is used widely for the engineering optimization to solve the complex problems and generation of the dynamic trees. The outcomes and results obtained and presented in this work is giving the variations in favor random forest based optimization for the software risk management and predictive mining. The need of the proposed work and background of the study includes the effective and performance based software bugs detection. The current problem addressed includes the accuracy and multi-dimensional evaluations. The key methodology adopted here to solve the existing problem is the integration of Random Forest approach and the findings are quite effective and cavernous in assorted aspects.

Index Terms: Artificial Neural Network, Random Forest Approach, Software Risk Management, Software Risk Prediction, Soft Computing for Software Bugs Prediction

I. INTRODUCTION

Software Risk Management involves the process of prior recognition and assessment of vulnerabilities with the classification approach so that the risk avoidance mechanism can be implemented. Software Risk Management is one the key factor in software project management with the goal to improve the quality as well as avoidance of vulnerabilities. The term Defect refers to the imperfection that may arise because of enormous reasons including programmers' skills, lack of suitable testing strategies and many others. When actual results are different from the expected result or meeting the wrong requirement is called as defect and it forms the basis of risk escalation in the software project, which is obviously not accepted in any type of deployment.

II. REVIEW OF RELATED WORK

M.Zavvar et al. (2017) focus on the parameters of Area Under Curve (AUC) and Classification Accuracy Rate (CAR) are evaluated and presented in this manuscript. The comparisons of the projected results are done with K-Means and Self Organizing Maps (SOM).

Edzreena Edza Odzaly et al. (2018) highlight the integration of data collection from live project environment so that the analytics on collected data can be done effectually.

Fuqiang Lu et al. (2017) empirically evaluate the performance states of GA, SA and SAGA with the overall conclusion that SAGA is quite effectual and performance aware as compared to the other traditional approaches. Daniel Mendez.

Fernandez et al. (2017) worked on the approach of evaluation of datasets in spreadsheets is integrated for the validation and final results.

Tobias Rauter et al. (2016) devised Asset-Centric based assessment of risk in the software components. The component-based model is adopted in this work with the identification of trust architecture for higher degree of security and overall integrity of the software development process.

Abdelrafe M. S. Elzamly (2016) presented 49 risk factors associated with the software project development and these can be effectually pushed back using fuzzy based approach with multiple regression.

Chaitanya Krishna et al. (2016) presented the evaluation and identification of software risks using Halstead approach so that inner factors, which are prominent for software, can be processed with the higher performance and maximum throughput.

Pradnya Purandare (2016) underlines the effectual and high performance application of Entropy in the management of risks in the software development process.

Morakot Choetkiertikul et al. (2015) predict the tasks and modules in the software projects, which can increase the risks of delays and additional time, which can be exploited as risk.

Abdelrafe Elzamly (2015) uses Linear Stepwise Discriminate Analysis (LSDA) Approach and techniques for the control of risks is also underlined and used so that the integrity of projected approach can be analyzed.

Ramakanta Mohanty et al. (2017) presented the work on machine learning based software defect prediction. The approaches used in the implementation include J48, GMDH, CART, TreeNet and Genetic Programming.

Revised Manuscript Received on June 15, 2019.

Alankrita Aggarwal, Department of Computer Science and Engineering, IKGPTU Jalandhar, Punjab, India.

Kanwalvir Singh Dhindsa, Department of Computer Science and Engineering, IKGPTU Jalandhar, Punjab, India.

P. K. Suri, Department of Computer Science and Engineering, Kurukshetra University, Haryana, India.

Retrieval Number: I11500789S19/19©BEIESP
DOI: 10.35940/ijitee.I1150.0789S19

Riya Singh et al. (2017) worked on the software defect prediction using averaging likelihood ensemble technique that is closely associated with the random forest algorithm. The benchmark dataset of PC4 is taken for research analytics and predictive mining.

Md.Mohsin Ali et al. (2017) presented the work on software defect prediction and metric selection using a parallel framework in the cloud environment.

Satya Srinivas Maddipati et al. (2018) integrated the adaptive neuro fuzzy inference system for software defects prediction. In this paper software defects are predicted using Adaptive Neuro Fuzzy Inference System (ANFIS).

Jifeng Xuan et al. (2017); Mamta Mittal et al., (2018) provides an approach to leveraging techniques on data processing to form reduced and high-quality bug data in software development and maintenance. In this paper, the authors combine feature selection with instance selection to reduce the scale of bug data sets as well as improve the data quality.

III. PROBLEM FORMULATION, PROPOSED APPROACH AND RESULTS

The present work is having key focuses on the development of a novel architecture and implementation using Random Forest Approach so that the higher degree of efficiency and accuracy can be achieved.

In case of Random forest Approach, there is generation of number of decision trees whereby the individual decision is associated with each tree. From all the possible permutations, the best outcome in terms of optimization is achieved. For this approach, the paradigm of voting is done in random forest approach.

To perform the simulation and implementation, a dataset from the real time software engineers are obtained and extracted so that the training of the bugs based records can be done. The data is inserted for training and further prediction in the following format.

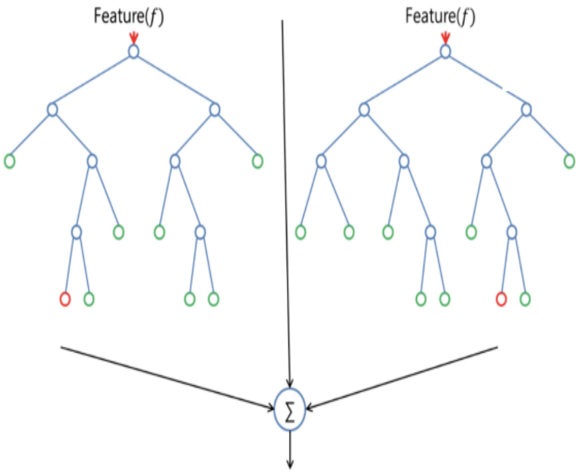


Fig. 1: Features in Random Forest

Figure 1 depicts the approach of RF with the tree formation in dynamic ways to have the outcomes for the decisions.

Table I: Format of the Test Dataset Used

Timestamp	Bug Id	Type of Bug	Penetration Level	Impact Type	Meta data

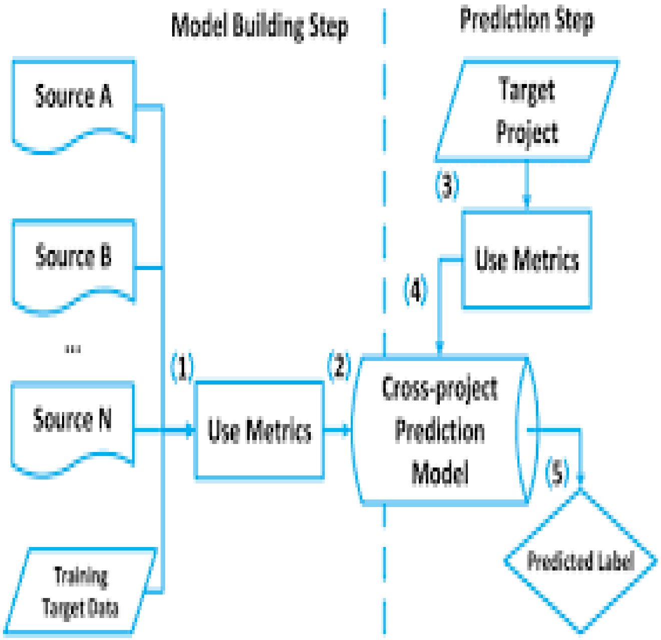


Figure 2: Model Building Approach for Predictions

The Figure 2 states the model building steps for the predictions .There are N number of inputs from various sources along with Training Target data. This input is using the metrics which in turn uses cross prediction model that uses metrics to target the project .The output we get is the data will be labeled as predicted. Prime Factors and Modules in the Model Presented are Cross Validation, Metrics and Perspectives, Labeling, Effective Training Data, Target Applications, Higher Degree of Validation Data.

From the model building approach, the metrics are used with the higher degree of performance in terms of the training of datasets with the getting of predictions values for the upcoming new datasets. The datasets of higher accuracy are presented so that the adaptability can be there with the insertion of the new datasets if required.

The term Defect alludes to the flaw that may emerge in light of colossal reasons including developers' aptitudes, absence of appropriate testing systems and numerous others. At the point when real outcomes are not the same as the normal outcome or meeting the wrong necessity is called as deformity and it shapes the premise of hazard heightening in the software venture which is clearly not acknowledged in a sending.

Following key parameters can be extracted for the simulation aspects i.e. Error Factor, Accuracy, Precision, Performance, Cumulative Effectiveness and Cost Factor.

A. Advantages of Random Forest Approach

- Generation of Huge Set of Solutions
- Each Set of Solution towards Optimization
- More options generation for the solutions spaces
- Each solution space is in optimized way for the engineering problems
- Effective outcomes for risk management

IV. RESEARCH METHODOLOGY

- Generation of Risk Datasets with Software Applications
- Key Extraction of Feature Points
- Thresholding with the Feature Points
- Generation and Spawning of Trees
- Each Branch association with the solutions
- Extraction of better outcomes and accuracy with random forest
- Usage of voting patterns for risk management

V. PROPOSED APPROACH WITH RANDOM FOREST ON DEFECTS AND BUGS

Spawning of the dataset with the generation of trees and branches towards the solutions is done so that the outcomes can be fetched on the base of the optimization and accuracy levels. The random forest approach with the ensemble-based learning is integrated with the decision outcomes for the effective solutions as in Figure 3.

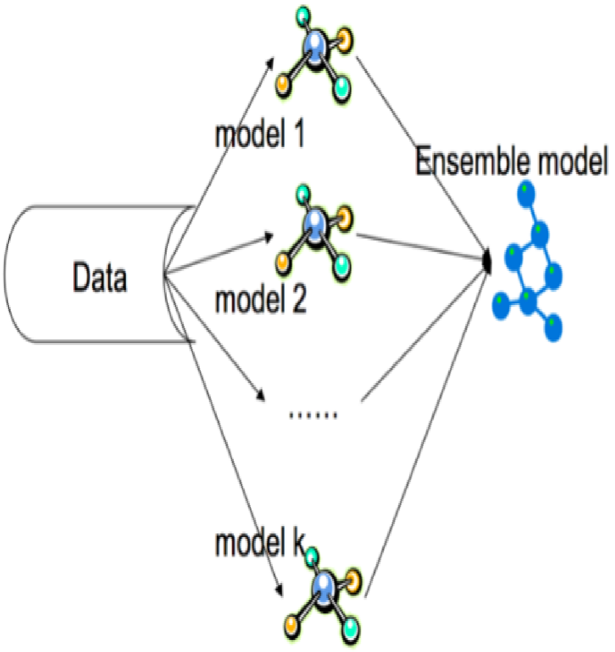


Fig. 3: Random Forest Approach with Ensemble Modeling

A. Properties of Random Forest Approach

Figure 4 is presenting the random forest approach and the instances with the spawning of trees and used in the software bugs prediction.

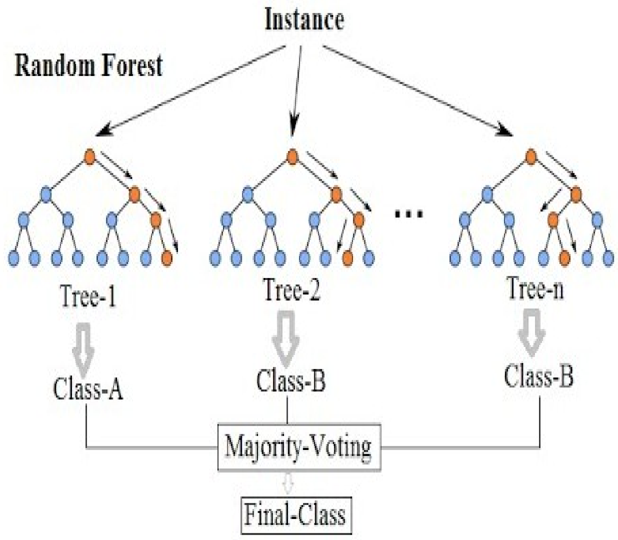


Fig. 4:Properties of Random Forest Approach

VI. IMPLEMENTATION OUTCOME

The implementation results are obtained from the model generated using Python libraries and found the effective outcomes. Python is one of the powerful programming languages for the high performance computing tasks and machine learning based predictions.

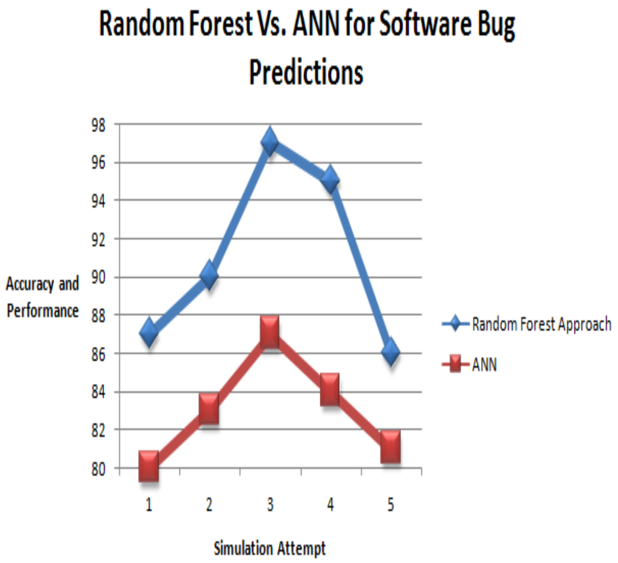
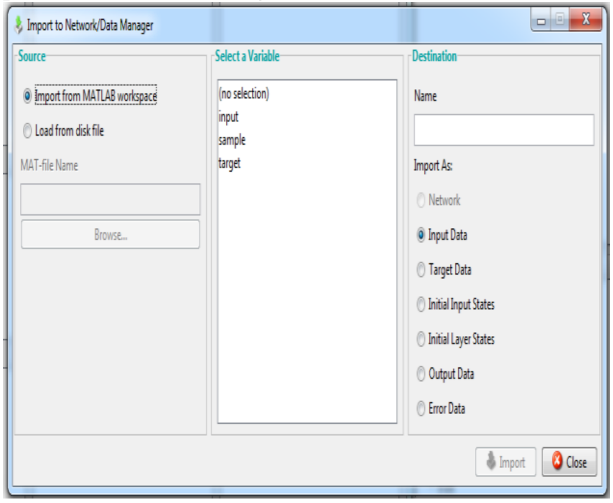


Fig. 5: Outcome and Predictive Values

As shown in Figure 5, on execution of the random forest approach and its comparative evaluation with traditional artificial neural network, it is found that the random forest based approach is quite better and effectual in terms of multiple parameters. This section presents the research design with the research foundations associated with the proposed approach. It includes research objectives and the assumptions of the study and research objectives and flow of approach. The segment puts forth details about its structure, the guidelines observed in designing the implementation perspectives along with the algorithmic approach.

VII. FINAL RESULTS AND SUMMARY

The results from ANN based model and training aspects are giving better and effective results in terms of software defects predictions shown in Figure 6.



Import to Network/Data Manager

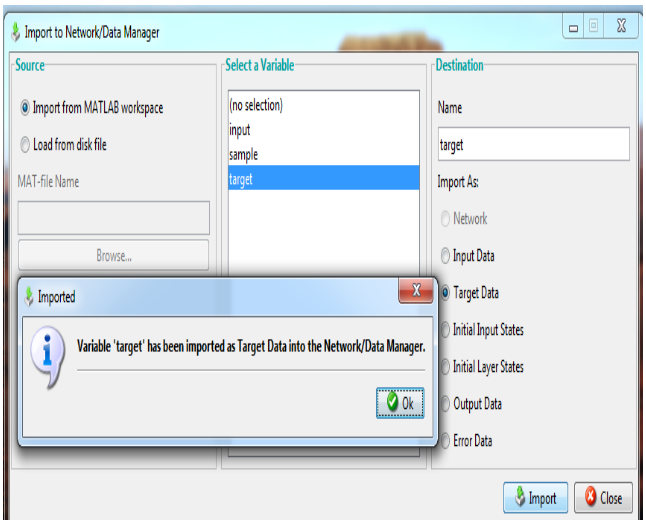
Source: ● Import from MATLAB workspace

Select a Variable: (no selection), input, sample, target

Destination – Import As: ● Input Data

Fig. 6: Data Fetching to MATLAB workspace

The dataset is read and extracted to the MATLAB workspace whereby the training is done with the outcomes on the objectives on consideration and presented in Figure 7.



Import to Network/Data Manager

Source: ● Import from MATLAB workspace

Select a Variable: target

Destination – Name: target

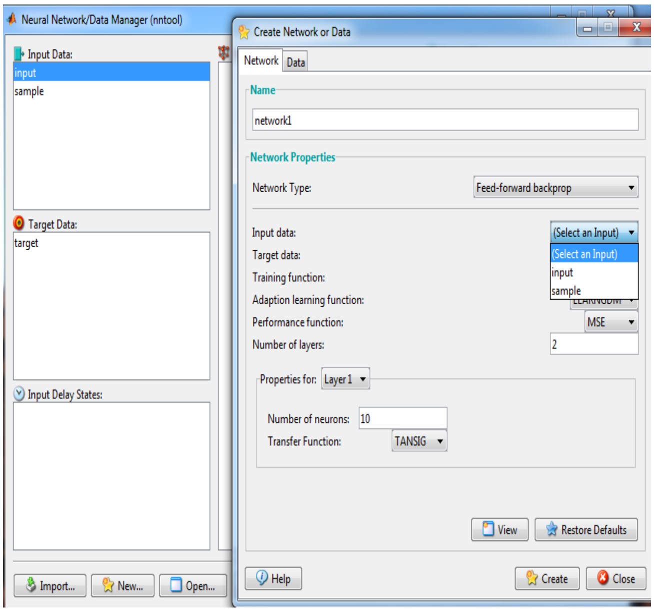
Import As: ● Target Data

Imported

Variable 'target' has been imported as Target Data into the Network/Data Manager.

Fig. 7: Successful import of variables (input, sample and output)

Figure 8 depicts the successful import of variables (input, sample and output) to the Working environment and further will be used for training and predictions.



Neural Network/Data Manager (nntool)

Input Data: input, sample

Target Data: target

Create Network or Data

Name: network1

Network Type: Feed-forward backprop

Input data: (Select an Input) – (Select an Input), input, sample

Performance function: MSE

Number of layers: 2

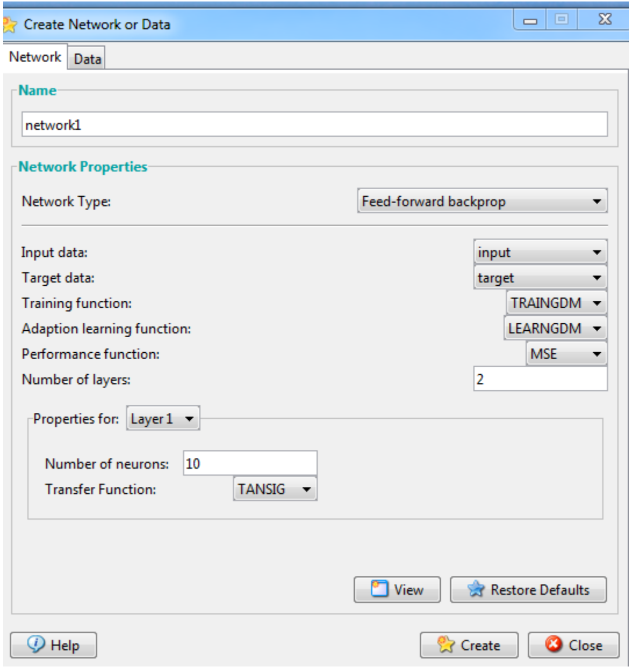
Properties for: Layer 1

Number of neurons: 10

Transfer Function: TANSIG

Fig. 8: New feedforward backpropagation network

The new feedforward network of type backpropagation approach is adopted for the training and further evaluations as in Figure 9.



Create Network or Data

Name: network1

Network Properties

Network Type: Feed-forward backprop

Input data: input

Target data: target

Training function: TRAINGDM

Adaption learning function: LEARNGDM

Performance function: MSE

Number of layers: 2

Properties for: Layer 1

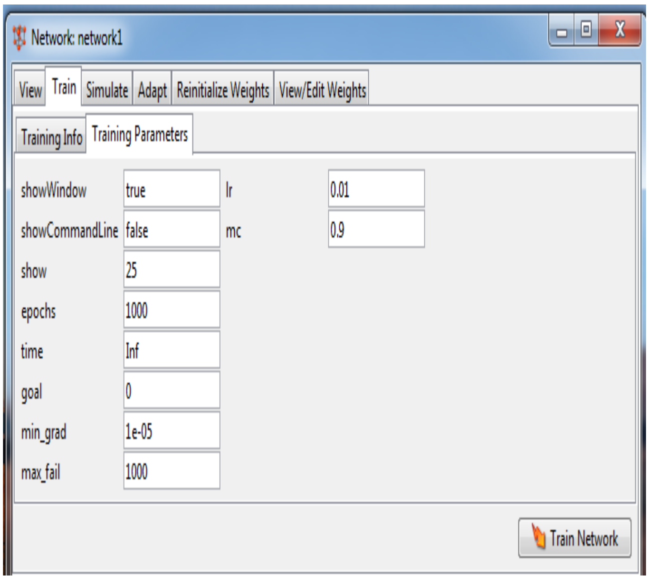
Number of neurons: 10

Transfer Function: TANSIG

Fig. 9: Setup of parameters for training

The setup of parameters involves the neurons or the data elements in process as shown in Figure 10.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication



showWindow	true	lr	0.01
showCommandLine	false	mc	0.9
show	25		
epochs	1000		
time	Inf		
goal	0		
min_grad	1e-05		
max_fail	1000		

Fig. 10: Determination of acceptability level and software defects anatomy.

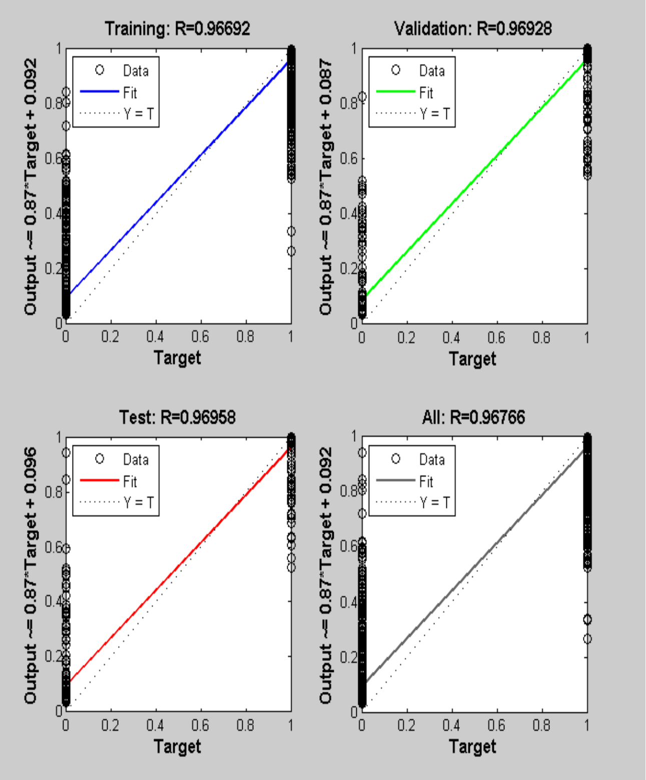


Fig. 11: Plotting of Residual Aspects

The different types of plots are required so that the residual values in addition to the final accuracy can be interrelated in Figure 11. In addition, the diversion of error factor should be towards minimum value in the plots. It provides the higher levels of accuracy in the cumulative outcomes.

VIII. CONCLUSION

The approaches associated with soft computing and machine learning are quite prominent and performance aware. In this paper, the usage of random forest approach is done so that the software risk predictions can be done with higher degree of performance. The procedure of Random Forest (RF) Approach is embraced in this work gives the viable results in a large portion of the cases when contrasted with ANN and along these lines the use examples of RF are execution mindful. The worldview of RF is utilized generally for the designing improvement to take care of the intricate issues and age of the dynamic trees. The results and results got and exhibited in this work is giving the varieties in support random forest based streamlining for the software risk management and prescient mining. Using random forest approach, the results found are quite effectual and towards the higher accuracy level on the datasets used at different instances.

ACKNOWLEDGMENT

The authors are thankful to IKGPTU Kapurthala Punjab (India) to facilitate us to pursue this research. Our sincere thanks also goes to the anonymous reviewers for their valuable remarks.

REFERENCES

[1] Yavari, M. Zavvar, S. M. Mirhassannia, M. R. Nehi, A. Yanpi, and M. H. Zavvar, “Classification of Risk in Software Development Projects using Support Vector Machine”, *Journal of Telecommunication Electronics Computer Engineering*, vol. 9(1), 2017, pp. 1–5.

[2] D. Greer, E. E. Odzaly, and D. Stewart, “Agile risk management using software agents”, *Journal of Ambient Intelligence and Humanized computing*, vol. 9(3), 2018, pp. 823-841.

[3] F. Lu, H. Bi, M. Huang, and S. Duan, “Simulated Annealing Genetic Algorithm Based Schedule Risk Management of IT Outsourcing Project”, *Mathematical Problems in Engineering,* 2017, pp. 1-17.

[4] M. Tießler, D. M. Fernández, M. Kalinowski, M. Felderer, and M. Kuhrmann, “On Evidence-based Risk Management in Requirements Engineering,”, *International Conference on Software Quality,* 2018, pp. 39–59.

[5] T. Rauter, N. Kajtazovic, and C. Kreiner , “Asset-Centric Security Risk Assessment of Software Components,” *2nd International workshop on MILS: Architecture and Assurance for Secure Systems*, 2016, pp. 1-17

[6] Abdelrafe MS, ”Managing Software Project Risks Using Stepwise and Fuzzy Regression Analysis Modeling Techniques”, 2016.

[7] C. Krishna and K. Subrahmanyam, “A Decision Support System for Assessing risk using Halstead approach and Principal Component Analysis,”, vol. 9(4), 2016, pp. 3383–3387.

[8] P. Purandare, “An entropy based approach for risk factor analysis in a software development project” *International Journal of Applied Engineering Res.*, vol. 11(4), 2016, pp. 2258–2262.

[9] M. Choetkiertikul, H. K. Dam, T. Tran, and A. Ghose, “Predicting delays in software projects using networked classification,”, *Proc. - 2015 30th IEEE/ACM International Conference of Automation Software engineering , ASE 2015*, 2016, pp. 353–364.

[10] Elzamly, B. Hussin, S. S. A. Naser, and M. Doheir, “Predicting Software Analysis Process Risks Using Linear Stepwise Discriminant Analysis: Statistical Methods,” *Int. J. Adv. Inf. Sci. Technol.*, vol. 38(38), 2015, pp. 108–115.

[11] Mohanty, R., & Ravi, V, “Machine Learning Techniques to Predict Software Defect. In Artificial Intelligence: Concepts, Methodologies, Tools, and Applications”, *IGI global*, 2017, pp. 1473-1487.

[12] Singh, R., Raja, R., & Chopra J, ” Software Defect Prediction Using Averaging Likelihood Ensemble Technique”, 2017

[13] Ali, M. M., Huda, S., Abawajy, J., Alyahya, S., Al-Dossari, H., & Yearwood, J. A, ” Parallel framework for software defect detection and metric selection on cloud computing”, *Cluster Computing*, vol. 20(3), 2017, pp. 2267-2281.

[14] Osman, H., Ghafari, M., & Nierstrasz, O, “Automatic feature selection by regularization to improve bug prediction accuracy.”, *In Machine Learning Techniques for Software Quality Evaluation (MaLTeSQuE), IEEE Workshop on IEEE,* 2017, pp. 27-32.

[15] Maddipati, S. S., Pradeepini, G., & Yesubabu, A, ” Software Defect Prediction using Adaptive Neuro Fuzzy Inference System.” *International Journal of Applied Engineering Research*, vol. 13(1), 2018, pp. 394-397.

[16] Lalit Mohan Goyal, Mamta Mittal, Iqbaldeep Kaur, Sumit Kaur, Amit Verma, D. Jude Hemanth, “Performance

Enhanced Growing Convolutional Neural Network Based Approach for Brain Tumor Segmentation in Magnetic Resonance Brain Images", *Applied Soft Computing, (SCIE Indexed journal), 4.004 (Second Revision Submitted)*

[17] Xuan, J., Jiang, H., Hu, Y., Ren, Z., Zou, W., Luo, Z., & Wu, X, "Towards Effective Bug Triage with Towards Effective Bug Triage with Software Data Reduction Techniques", *arXiv preprint,* 2017. arXiv: 1704.04761

AUTHORS PROFILE

Alankrita Aggarwal is a PhD research scholar from IK Gujral Punjab Technical University Jallandhar-144603, Punjab (India). She is M.tech and BE in Computer science and engineering from Maharishi Dayanand University ,Rohtak (India) having a rich experience approx 15 years of teaching as well as research experience .Her areas of interest are soft computing, software engineering and neural networks. She has published 22 research papers in reputed International journals and guided more than 12 M.tech dissertations. She also worked as the organizing secretary and coordinator of various Workshops. She is a Member of International Associations of Engineers (IAENG) also lifetime member of ISTE.

Dr. Kanwalvir Singh Dhindsa is professor in computer science and engineering department at Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib-140407,Punjab(India) affiliated to IKGPTU Jallandhar Punjab (India).He is having more than 20 years of Industrial ,teaching as well as research experience. He has published research papers in reputed International journals and guided Phd as well as M.tech students. His areas of interest are cloud computing, Big data, IoT, Mobile Computing, Database & security, Web Engineering. He is also member of Computer Society of India (CSI), Indian Society of Technical education (ISTE), Institution of Engineers (IEI).He is also convener of various national as well international conferences and workshops.

Dr.P.K.Suri is a former Professor, Dean Academic and Chairman of department of computer science and applications, Kurukshetra University, Kurukshetra-136119,Haryana,India and HCTM Technical Campus, Kaithal (Haryana), India. He has forty years of teaching and research experience with various designations in both the institutes. He received his MSc Degree from IIT Roorkee (formerly known as University of Roorkee) in the year 1972.He has completed hid Phd Degree at Faculty of Engineering, Kurukshetra University ,Kurukshetra in year 1981.His research interests include simulation, cloud computing, Adhoc networks, wireless networks, distributed systems, software engineering. He has attended number of national and international conferences and published a number of research papers in national and international journals. He has guided more than 20 Phd Research scholars.