ISSN 2348 – 8034
Impact Factor- 4.022

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

# ROLE OF DATA MINING CLASSIFICATION TECHNIQUE IN SOFTWARE DEFECT PREDICTION

**Dr.A.R.Pon Periyasamy[*1] and Mrs A.Misbahulhuda[2]**

[*1]Associate Professor, Dept. of Computer Science, Nehru Memorial College, Puthanampatti. Tamilnadu, India- 621 007

[2]Research Scholar, Dept. of Computer Science, Nehru Memorial College, Puthanampatti. Tamilnadu, India- 621 007

## ABSTRACT

Software defect prediction is the process of locating defective modules in software. Software quality may be a field of study and apply that describes the fascinating attributes of software package product. The performance should be excellent with none defects. Software quality metrics are a set of software package metrics that target the standard aspects of the product, process, and project. The software package defect prediction model helps in early detection of defects and contributes to their economical removal and manufacturing a top quality software package supported many metrics. The most objective of paper is to assist developers determine defects supported existing software package metrics victimization data mining techniques and thereby improve the software package quality. In this paper, role of various classification techniques in software defect prediction process are analyzed.

***Keywords***: *Classification Techniques, Data Mining, Defect Prediction, Software Quality.*

## I. INTRODUCTION

Now a days, the researchers have found that the quality of software datasets had serious effect on the performance of predicting software faults. In context of software package engineering, software package quality refers to software package purposeful quality and software package structural quality. Software package practical quality reflects useful needs whereas structural quality highlights non-functional needs. Software package metrics specialize in the standard side of the merchandise, method and project. During this paper the most stress is on software package. The target of software package quality engineering is to realize the desired quality of the product through the definition of quality needs and their implementation, activity of acceptable quality attributes and analysis of the ensuing quality .Software quality measuring is regarding quantifying to what extent a system or software package possesses fascinating characteristics specifically responsibility, Efficiency, Security, Maintainability and (adequate) Size. This may be performed through qualitative or quantitative means that or a mixture of each. In each cases, for every fascinating characteristic, there are a collection of measurable attributes like Application design Standards, coding Practices, Complexity, Documentation, movability and Technical &amp; practical volumes. The existence of those attributes in an exceedingly piece of software package or system tends to be correlate and related to this characteristic [10], [14], [25]. [38], [39], [41], [45].

Kamei and Shihab suggest that the NASA datasets remain the most popular for defect prediction, and also report that the PROMISE repository is used increasingly. Ease of availability mean that these datasets remain popular despite reported issues of data quality[50 ]. Divya Tomar and Sonali Agarwal have presented a software defect prediction system using Weighted Least Squares Twin Support Vector Machine (WLSTSVM). This system assigns higher misclassification cost to the data samples of defective classes and lower cost to the data samples of nondefective classes. The experiments on eight software defect prediction datasets have proved the validity of the proposed defect prediction system. The significance of the results has been tested via statistical analysis performed by using nonparametric Wilcoxon signed rank test[51]. M. Jaikumar, A. V. Ramani have offered the detailed survey regarding Software Defect Prediction techniques along with taxonomy of literatures [52]. David Bowes, Tracy Hall, Jean Petric have conducted a sensitivity analysis to compare the performance of Random Forest, Na¨ıve Bayes, RPart and SVM classifiers when predicting defects in NASA, open source and commercial datasets. The defect predictions that each classifier makes is captured in a confusion matrix and the prediction uncertainty of each classifier is compared. Despite similar predictive performance values for these four classifiers, each detects different sets of defects. Some classifiers are more consistent in predicting defects than others[53].

## II. SOFTWARE DEFECT PREDICTION-APPROACHES AND METHODOLOGIES

A software package defect is miscalculation, flaw, failure, or fault during an exceedingly computer program or system that causes it to provide an incorrect or sudden result, or to behave in unplanned ways that. Most defects arise from mistakes and errors created by individuals in either a program's source code or its style, or in frameworks and operative systems utilized by such programs, and some are caused by compilers manufacturing incorrect code.Software Defect Prediction Model refers to those models that attempt to predict potential software package defects from check information. There exists a correlation between the software package metrics and also the fault disposition of the software package. A software package defect prediction models consists of independent variables (Software metrics) collected and measured throughout software package development life cycle and variable (faulty or non-faulty). There are completely different data mining techniques for defect prediction.

Data mining is playing vital role in prediction of software defects. Data mining is a process of data analysis from various perspectives and summarizes it into useful information. It helps users to understand the substance of the relationships between the data. Data mining is that the analysis step of the "Knowledge Discovery in Databases" method, or KDD, a method of discovering patterns in massive data sets involving strategies at the intersection of computer science, machine learning, statistics, and database systems. The goal of data mining method is to extract information from an information set and rework it into a plain structure for more analysis[2], [3], [29], [34], [37], [44].

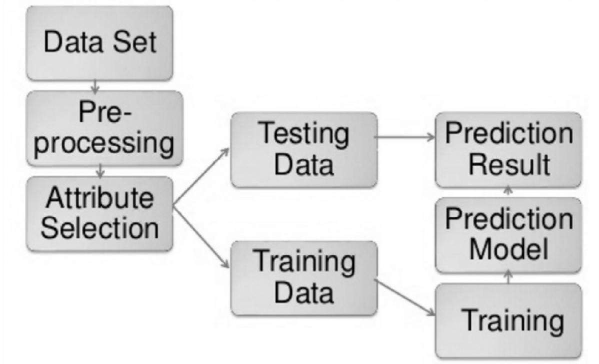

*Figure 1: Software Defect Prediction Model*

Data Mining will be divided into 2 tasks: prognosticative tasks and descriptive tasks. Prognosticative task is to predict the worth of a selected attribute (target/dependent variable) based on the worth of alternative attributes (explanatory). Descriptive task is to derive patterns (correlation, trends, and trajectories) that summarize the underlying relationship between information.

There are numerous data mining techniques used for software package defect predictions that are mentioned below.

*1. Regression:* it's a statistical method to judge the connection among variables. It analyses the link between the dependent or response variable and freelance or predictor variables. The connection is expressed within the kind of an equation that predicts the response variable as a linear operate of variable quantity. [5],[49].

*2. Association Rule Mining:* it's a technique for locating fascinating relationships between variables in massive databases. It's regarding finding association or correlations among sets of things or objects in database. It essentially deals with finding rules that may predict the prevalence of item supported the prevalence of alternative things [12], [33].

*3. Clustering:* Clustering could be a way to categories' a set of things into groups or clusters whose members are similar in a way. it's task of grouping a group in such the simplest way that items within the same cluster area unit the same as alternative and dissimilar to those in other clusters [17], [20].

*4. Classification:* It consists of predicting a particular outcome supported a given input. Classification technique use input file, additionally referred to as training set wherever all objects are already labeled with known category labels. The target of classification algorithm is to research and learns from the training data set and develop a model. This model is then wont to classify check data that the category labels aren't known [4], [7], [11], [21], [22],[23], [30], [35], [47]. The assorted classification techniques are given below.

*a.Neural Networks:* Neural Networks are the non linear prognosticative models which may learn through training and correspond biological neural networks in structure. A neural network consists of interconnected process components known as neurons that employment along in parallel among a network to supply output. [19], [28], [48].
*b. Decision Trees:* a choice tree could be a prognosticative model which may be accustomed represent each classification and regression models within the kind a tree structure. It refers to a hierarchical model of choices and their consequences. It's a tree with decision nodes and leaf nodes. A decision node has 2 or a lot of branches. Leaf nodes represent a classification or decision [26], [40].

*C.Naive Bayes:* it's supported Bayes theorem with independence assumption between predictors. Naive Bayes Classifier is predicated on the belief that the presence or absence of a specific feature of a category in not associated with the presence or absence of the other feature [13], [27], [32],[36], [42], [46].

*d.Support Vector Machines:* SVM are supported the construct of decision planes that outline decision boundaries. a decision plane is that the one that separates between a collection of objects having completely different category membership. SVM is primarily a classifier methodology that performs classification task by constructing hyper plane during a three-dimensional area that separates cases of various class labels. It supports each regression and classification [1], [6], [8].

*e.Case based Reasoning:* Case based reasoning suggests that determination new issues supported the similar past issues and victimization recent cases to clarify new things. It works by comparison new unclassified records with known examples and patterns. an easy example of a case based mostly learning algorithm is k-nearest neighbor algorithm. It's simple algorithm that stores all on the market cases and classifies new cases supported a similarity live i.e. distance operate. [9], [19]

Table1 shows the comparative analysis of Algorithms for supervised Classification kind.

***Table 1: Comparative Analysis of Supervised Classification Type Dataset***

| Algorithm | Pros | Cons |
|---|---|---|
| **BR** | Fits Calculation diagonal matrices | No tag correlations performed explicitly |

| | | |
|---|---|---|
| **Ada boost** | Excellent for sorting better accuracy | Generalizing results in decreased performance |
| **Back Propagation** | Learning iteratively. More capacity generalization | Computationally complex presented by the algorithm |
| **C4.5** | Based on decision trees, improving accuracy and prediction. Easy to understand, popular and powerful | Not takes correlation between classes |

## III. Software Defect Prediction (SDP) victimization completely different Classification Techniques

A survey is conducted to assist developers determine defects supported existing software package metrics victimization data mining techniques particularly Classification and there by improve software package quality that ends up in reduction within the software package development value within the development and maintenance section. Different classification techniques are surveyed with completely different data sets.

### *3.1 SDP victimization supervised Learning*

The various supervised Learning techniques are mentioned during this section.

### *3.1.1 SDP victimization Bayesian Network*

Yuan Chen, et.al [30] have surveyed the various data mining classification techniques for software package defect prediction. They projected a replacement model based on Bayesian network and PRM to predict the software package defect and manage. Hassan Najadat and Izzat Alsmadi [16]Proposed a replacement model supported Ridor algorithm to predict fault in modules. They additionally tested the various classification techniques on the data sets provided by NASA. The results shown that Ridor algorithm is best than the present technique in terms of accuracy and extraction of variety of rules. Ahmet Okutan,Olcay Taner Yıldız [42],Introduced a replacement two metrics NOD, for the amount of developers and LOCQ for source code quality excluding the metrics that is accessible in Promise knowledge repository. Using Bayesian network classifier experimental shows that noc &DIT have terribly restricted and untrustworthy. LOCQ is more practical like CBO &amp; WMC. NOD metric showed that there's a direct correlation between the no of developers and extent of defect prunes. LOC is established to be one amongst the most effective metric for fast defect prediction. LCOM3 &amp; LCOM have less effective compared to LOC, CBO, RFC, and LOCQ, WMC. Thair nu Phyu [7] reviewed on numerous classification techniques like decision tree induction, Bayesian networks, k-nearest neighbor classifier, case-based reasoning,

*Table 1: Table 2. Comparative Analysis of Semi-Supervised*

| Algorithm | Pros | Cons |
|---|---|---|
| Multi-label classification by constrained non-negative matrix factorization | Adaptable to semi-supervised environments along with the representation of documents in rank matrix factorization<br>Using the representation of documents in rank matrix factorization using the non-negative. | There is a strong influence from two parameters on the performance: latent variables and tuning<br>Parameters. |
| **Graph-based SSL with multi-label** | Effective use of large amounts of unlabeled data and the ability to exploit the relationships between labels | Most of the time is used for video files. It does not adapt well to texts. |
| Multi-label learning by using dependency among labels | Improving accuracy by configuring SSL | Time increment for large data sets |
| Semi-supervised multi-label learning by solving a Sylvester | Use of large amounts of unlabeled data as well as the ability to exploit the relationship between labels. Significant improvement in the precision. | May become slow when using large data sets |

| | | |
|---|---|---|
| **Semi-supervised non-negative matrix factorization** | Using NMF in conjunction with SSL allows the extraction of the most discriminating than if MFN were used. | Computational Complexity |

### *3.1.3. SDP exploitation Support Vector Machine*

Sonali Agarwal and DivyaTomar [31] have planned a feature choice primarily based Linear Twin Support Vector Machine (LSTSVM) model to predict defect prone software package modules. F-score technique is employed for software package defect prediction supported numerous software package metrics. This model is applied on PROMISE data sets and compared with the opposite existing models. The results say that the performance of the new model is best than the present machine learning models.CagatayCatal [18] planned four semi-supervised classification strategies like Low-density separation (LDS), support vector machine (SVM), expectation-maximization (EM-SEMI), and class mass normalisation (CMN) for semi-supervised defect prediction. They applied four kinds of ssc on nasa datasets. The results showed that SVM &amp; LDS are higher than CMN and EM-SEMI. LDS performs far better than SVM for an oversized data set.

Karim O. Elish, Mahmoud OElish [6] planned SVM is that the model and compared with the eight completely different statistical and Machine learning models The compared models are 2 applied math classifiers techniques: (I)Logistic Regression (LR) and (ii) K-Nearest Neighbour (KNN); 2 neural networks techniques: (I) Multi-layer Perceptrons (MLP) and (ii) Radial Basis Function(RBF); two Bayesian techniques: (I) Bayesian Belief Networks (BBN) and (ii) Naı¨ve Bayes (NB); and 2 tree structured classifiers techniques: (I) Random Forests (RF) and (ii) decision Trees (DT) victimization four nasa information sets. The results found that SVM is that the higher model in comparison to the opposite models. David Grayet.al [8] planned a piece victimization the static code metrics for a set of modules contained inside eleven nasa information sets are used with a Support Vector Machine classifier. A rigorous sequence of pre-processing steps were applied to the information before classification, as well as the leveling of each categories (defective or otherwise) and also the removal of an oversized variety of repetition instances. The Support Vector Machine during this experiment yields a mean accuracy of seventieth on antecedently unseen information.

### *3.1.4. SDP victimization decision Tree*

GolnoushAbaei•AliSelamat [41], during this paper many various machine learning techniques like decision trees, decision tables, random forest, neural network, Naïve Bayes and distinctive classifiers of artificial immune systems (AISs) like artificial immune recognition system, CLONALG and Immunos. Experiment is performed on four public nasa informationsets that are totally different in size and range of defective data. The results obtained are random forest provides the most effective prediction performance for big data sets and Naïve Bayes may be a trustable algorithm for little information sets even once one in all the feature choice techniques is applied. Immunos99 performs well among AIS classifiers once feature choice technique is applied, and AIRS Parallel perform higher with none feature choice techniques. Thair nu Phyu [7] reviewed on numerous classification techniques like decision tree induction, Bayesian networks, k-nearest neighbour classifier, case-based reasoning, genetic rule and fuzzy logic techniques. The results found that there's no correct data that that is that the best classifier. Many of the classification strategies turn out a collection of interacting loci that best predict the constitution. However, a simple application of classification strategies to giant numbers of markers includes a potential risk memorizing indiscriminately associated markers.

### *3.2 SDP using Semi-supervised Learning*

Ming Li, et al.proposed [15] a sample based mostly strategies for software package defect prediction. 3 strategies like sampling with standard machine learners, sampling with a semi-supervised learner and active sampling with active semi-supervised learner. They applied a semi-supervised learning methodology known as ACoForest to create a classification model supported a sample conjointly the remaining un-sampled mod-ulesthey also projected a unique active semi supervised methodology known as AcoForest which might choose unsampled modules and experimented on Promise data sets and located to be the simplest technique. Experimental results show that size doesn't have an effect on the defect prediction. CagatayCatal [18] planned four semi-supervised classification strategies like Low-density separation (LDS), support vector machine (SVM), expectation-maximization (EM-SEMI), and class mass standardisation (CMN) for semi-supervised defect prediction. They applied four kinds of ssc on NASA datasets. The results showed that SVM &amp; LDS are higher than CMN and EM-SEMI. LDS performs far better than SVM for a large data set. G.Abaeia, A.Selamata, H.Fujitab have conducted a study based on semi-supervised hybrid self-organizing map for software fault prediction [43]..

### *3.3. SDP using unsupervised Learning*

C.Chung and S.Dhall[24] projected a numerous classification and cluster strategies to predict software package defect. the assorted data mining classifier algorithms specifically J48, Random Forest, and Naive Bayesian Classifier (NBC) are evaluated supported numerous criteria like roc, Precision, MAE, RAE etc. cluster technique is later applied on totally different data set of NASA victimization k-means, hierarchical cluster and create Density based mostly cluster algorithm. Results are evaluated supported criteria like Time Taken, Cluster Instance, range of Iterations, Incorrectly Clustered Instance and Log probability etc. Dhiman,et al. [14] projected a model wherever in it'll reason the software package defects victimization some cluster approach then the software package defects are measured in every clustered individually. this method can analyze the software package defect and its integration with software package module.

### *3.4. SDP using Machine Learning algorithm*

Xiao-Yuan Fing,et.al [30] have tried to model the effective , economical and low procedure burden victimization advanced machine learning technique like cooperative representative classification. The new model projected by them is CSDP that is employed to predict defect terribly very economical manner. Kehan Gao&amp;Taghi M [11] experimented on promise repository supported criteria i) Feature choice supported sampled data, and modelling supported original data, ii) Feature choice based on sampled data modelling supported sampled information and iii) Feature choice supported sampled information, and modelling based mostly sample data. The experimental results showed that the first criteria is that the best compared to the others in defect prediction. S.Bibiet al [5] planned a RVC model for locating the defects within the software package by victimization symbolic learning algorithms. they need compared the model with many machine learning algorithms in 2 software package data sets and also the results found were higher regression error than the quality regression approaches on each information sets.

## IV. CONCLUSION

Software quality is that the degree of conformity to express or implicit necessities and expectations. A software package metric could be a quantitative live of a degree to that a software package or method possesses property with no defects. Hence, software package defect prediction model helps in early detection of defects victimization Classification Technique. During this paper we've got mentioned the varied classification techniques like supervised, Un-supervised and Semi-supervised that are applied on numerous datasets supported existing software package metrics. In future we'll be comparison the results of Supervised classification techniques on completely different datasets and open source comes to research the most effective classification technique to predict the defect so as to evolve an honest software package quality product.

## REFERENCES

1. G. H. Jozsefvalyon, "Least Squares Support Vector Machines for Data Mining", Budapest University of Technology and Economics, Department of Measurement and Information Systems, published in Neural Networks, Proceedings, IEEE International Joint Conference, (2003).

2. Y. Ma,C. Bojan,"Singh:Robust prediction of fault-proneness by random forests ,Software Reliability Engineering", ISSRE 2004. 15th International Symposium,(2004),pp. 417-428.

3. E. O. Costa, G. A. de Souza, A. T. R.Pozo, and S. R.Vergilio, "Exploring Genetic Programming and Boosting Techniques to Model Software Reliability", IEEE Transaction on Reliability, vol. 56, no. 3, (2007).

4. S.Lessmann,B.Baesens,C.Mues,and S. Pietsch,"Benchmarking Classification Models for Software Defect Prediction: A Proposed Framework and Novel Findings", IEEE Transactions on Software Engineering, (2008).

5. S. Bibi,, G. Tsoumakas, I. Stamelos, I. Vlahavas, "Regression via Classification applied on software defect estimation",Elsiever,vol. 34, no. 3,(2008), pp. 2091-2101.

6. K. O. Elish, M. O. Elish, "Predicting defect-prone software modules using support vector machines", Elsevier, vol. 81, no. 5, (2008).

7. T. Nu Phyu, "Survey of Classification Techniques in DataMining", International MultiConference of Engineers and Computer Scientists, (2009); Hong Kong.

8. D.Gray,D. Bowes, N. Davey, Y. Sun, "Bruce Christianson, Using the Support Vector Machine as a Classification Method for Software Defect Prediction with Static Code Metrics",11th International Conference, EANN 2009, (2009); London, UK.

9. Y. Chen, P. Du,Xi , X.-H. Shen, "Research on Software Defect Prediction Based on Data Mining", Computer and Automation Engineering (ICCAE), 2nd International Conference, (2010), vol. 1, pp. 563-567.

10. M. Jureczko, "Significance of Different Software Metrics in Defect Prediction", Software Engineering An International Journal , vol. 1, no 1,2011, pp.86-95.

11. K. Gao, T..M.Khoshgoftarr, "Software Defect Prediction for high- dimensional and class-imbalanced data", 23rd International Conference on Software Engineering & Knowledge Engineering (SEKE'2011), Eden Roc Renaissance, (2011)Miami Beach, USA.

12. B. Ma, D. Karel, V. Jan, B. Bart, "Software defect prediction based on association rule classification", Research Center for Management Informatics (LIRIS), Leuven, (2011).

13. C. Catal, U.Sevim, B. Diri,"Practical development of an Eclipse-based software fault prediction tool using Naive Bayes algorithm", Elsevier,(2011).

14. P.Dhiman, M.C. Manish,"A Clustered Approach to Analyze the Software Quality Using Software Defects, Advanced Computing & Communication Technologies (ACCT)", 2012 Second International Conference,(2012).

15. M. L., H. Zhang, R. Wu, Z.-H. Zhou, "Sample-based software defect prediction with active and semi-supervised learning", Automated Software Engineering , (2012), vol. 19, no. 2, pp. 201-230

**16.** *H.Najadat and I.Alsmadi, "Enhance Rule Based Detection for Software Fault Prone Modules", International Journal of Software Engineering and Its Applications, vol. 6, no. 1, (2012).*

**17.** *S. Kaur, and D. 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,(2012).*

**18.** *C. Catal, "A Comparison of Semi-Supervised Classification Approaches for Software Defect Prediction", Journal of Intelligent Systems, vol. 23, no. 1, pp. 75-82,(2013).*

**19.** *R.Goyala, P.Chandraa, Y. Singha, "Suitability of KNN Regression in the Development of Interaction Based Software Fault Prediction Models", IERI Procedia, International Conference on Future Software Engineering and Multimedia Engineering, Elsiever, vol 6, pp. 15-21, (2013),.*

**20.** *G.Scanniello, C.Gravino, A.Marcus,T.Menzies,"Class level fault prediction using software clustering, Automated Software Engineering (ASE)", 2013 IEEE/ACM 28th International Conference, (2013).*

**21.** *B. V. Balaji1, V.Venkateswara Rao2, "Improved Classification Based Association Rule Mining", International Journal of Advanced Research in Computer and communication Engineering, vol. 2, no. 5, (2013).*

**22.** *R. M. Rahman, F. Afroz,"Comparison of Various Classification Techniques Using Different Data Mining Tools for Diabetes Diagnosis",Journal of Software Engineering and Applications, (2013), vol.6, pp.85-97.*

**23.** *T. Angel Thankachan1, K. Raimond, "A Survey on Classification and Rule Extraction Techniques for Data mining", IOSR Journal of Computer Engineering ,vol. 8, no. 5,(2013), pp. 75-78.*

**24.** *A. Chug1 and S. Dhall1, "Software Defect Prediction Using Supervised Learning Algorithm and Unsupervised Learning Algorithm",The Next Generation Information Technology Summit (4th International Conference),(2013),pp.1-6.*

**25.** *Anuradha chug, Shafali Dhall, "Software defect prediction using supervised learning algorithm and unsupervised learning algorithm", Confluence 2013: The Next Generation Information Technology Summit (4th International Conference), (2013).*

**26.** *M. Surendra Naidu, "Classification of Defects in Software Using Decision Tree Algorithm", International Journal of Engineering Science and Technology (IJEST), (2013).*

**27.** *A.TosunMisirli, A. se Ba¸ S.Bener,"A Mapping Study on Bayesian Networks for Software Quality Prediction", Proceedings of the 3rd International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering, (2014).*

**28.** *R.Kalsoom, M. Qureshi, "Application and Verification of Algorithm Learning Based Neural Network",arXiv preprint arXiv:1406.2614, (2014), arxiv.org.*

**29.** *A. Kaur and I. Kaur, "Empirical Evaluation of Machine Learning Algorithms for Fault Prediction", LectureNotes on Software Engineering, vol. 2, no. 2, (2014).*

**30.** *X. Yuan, H.W. Zhang, S. Ying,F. Wang, "Software defect prediction based on collaborative representation classification", Proceedings in ICSE Companion 2014, 36th International Conference on Software Engineering, pp. 632-633.*

**31.** *S. Agarwal and D.Tomar, "A Feature Selection Based Model for Software Defect Prediction", International Journal of Advanced Science and Technology, vol.65,(2014), pp. 39-58.*

**32.** *K. Sankar, S. Kannan and P.Jennifer, "Prediction of Code Fault Using Naive Bayes and SVM Classifiers Middle-East Journal of Scientific Research", vol. 20, no. 1, (2014), pp.108-113.*

**33.** *G.Czibula, Z. Marian, I. G.Czibula, "Software defect prediction using relational association rule mining, Information Sciences", vol. 264, no. 20 (2014), pp. 260-278.*

**34.** *R. Li, S.Wang, "Ann Empirical Study for Software Fault-Proneness Prediction with Ensemble Learning Models on Imbalanced Data Sets", Journal of Software, vol. 9, no.3,pp. 697-704,(2014).*

**35.** *M. Barcelo-Valenzuela, M. Romero-Ochaoa, A. Perez-Soltero, G. Sanchez-Schmitz, "Knowledge Sources and Automatic Classification: A Literature Review", International Journal of Business, Humanities and Technology, vol. 4, no. 1, (2014).*

**36.** *L. Li, H. Leung, "Bayesian Prediction of Fault-Proneness of Agile-Developed Object-Oriented System:Lecture Notes", Business Information Processing, vol. 190, (2014), pp. 209-225.*

**37.** *The Global Conference for Wikimedia,(2014); London.*

**38.** *L. Madeyski, M.Jureczko, "Which process metrics can significantly improve defect prediction models?", An empirical study,(2014).*

**39.** *D.Mehta, "A Comparative study of Techniques in Data Mining", by Manika Verma1, International Journal of Emerging Technology and Advanced Engineering, vol. 4, no. 4, (2014).*

**40.** *P. Reena, R. Binu, "Software Defect Prediction System –Decision Tree Algorithm with Two Level Data Pre-processing", International Journal of Engineering Research & Technology (IJERT), vol. 3, no. 3, (2014).*

**41.** *G.Abaei, A.Selamat, "A survey on software fault detection based on different prediction approaches", Vietnam Journal of Computer Science, (2014), vol. 1, no. 2, pp. 79-95.*

**42.** *A.Okutan, O. T.Yildiz, "Software defect prediction using Bayesian networks", Empirical Software Engineering, (2014), vol. 19, no. 1, pp. 154-181.*

**43.** *G.Abaeia, A.Selamata, H.Fujitab, "An empirical study based on semi-supervised hybrid self-organizing map for software fault prediction", Knowledge-Based Systems, vol. 74, (2015), pp. 28-39.*

**44.** *R. Malhotra, "A systematic review of machine learning techniques for software fault prediction",Applied Soft Computing, vol. 27, (2015), pp. 504-518.*

**45.** *H. Laradji, M.Alshayeb, L.Ghouti, "Software defect prediction using ensemble learning on selected features. Information and Science Technology", vol. 58, (2015), pp. 388-402.*

**46.** *W. Zhang, Y. Yang, Q. Wang, "Using Bayesian Regression and EM algorithm with missing handling for software effort prediction", Information and software technology, vol. 58, (2015), pp. 58-70.*

47. Ajay Prakash, D. V. Ashoka, V. N. ManjunathAradya, "Application of Data Mining Techniques for Defect Detection and Classification", Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014, Advances in Intelligent Systems and Computing, vol. 327, (2015), pp. 387-395

48. G.Bhavya, "Improving the Fault Prediction in OO Systems Using ANN with Firefly Algorithm",International Journal of Innovative Research in Science & Engineering, pp. 2347-3207.

49. S.Bibi, G.Tsoumakas, I.Stamelos, I. Vlahavas, "Software Defect Prediction Using Regression via Classification", Department of Informatics, Aristotle University of Thessaloniki,54124 Thessaloniki, Greece.

50. Kamei, Y., & Shihab, E. (2016). Defect prediction: Accomplishments and future challenges. In 2016 IEEE 23rd international conference on software analysis, evolution, and reengineering , vol 5, pp33–45.

51. Divya Tomar and Sonali Agarwal, "Prediction of Defective Software Modules Using Class Imbalance Learning", Applied Computational Intelligence and Soft Computing, 2016.

52. M. Jaikumar, A. V. Ramani, " Software Defect Prediction–Taxonomy of Literatures and Research Dimensions", International Journal of Engineering Science and Computing, May 2016, Vol 6, No. 5, pp 6157-6160.

53. David Bowes, Tracy Hall, Jean Petric, "Software defect prediction: do different classifiers find the same defects?", Software Qual J, Springer 2017.