TEST Engineering & Management

March - April 2020
ISSN: 0193-4120 Page No. 2245 - 2253

Feature Selection Techniques for Software Defect Prediction : A Literature Review

Varsha Palatse
Dept. of Computer Engineering
Government Polytechnic Pune,
Maharashtra, India
varshapalatse@gmail.com

Article Info
Volume 83
Page Number: 2245 - 2253
Publication Issue:
March - April 2020

Article History
***Article Received**: 24 July 2019*
***Revised**: 12 September 2019*
***Accepted**: 15 February 2020*
***Publication**: 18 March 2020*

Abstract
Software defect prediction plays a vital role to identify the most defect prone modules or components of software. Its aim is to determine software reliability via learning from historical defect data. Feature selection is used to train the prediction models. It helps to enhance the performance of prediction and reduce the computation time of models. Various studies have been carried out on feature selection methods within the project or cross projects. The purpose of this work is to synthesize the literature of previous studies on different feature selection techniques with respect to small size of metrics data in different context , to find out the choice of training data, and the modelling techniques applied that have great impact on the performance of prediction models. We have conducted the literature review to evaluate the feature selection methods proposed from 2009 to 2019.The results are analysed qualitatively and quantitatively of 24 studies. It reveals approaches and adequate context-specific used information based on the criteria we construct and apply. The discussions are identified from 24 studies and analysed by considering the assessment points. As per the literature review feature selection techniques performed well , contributed to allocate minimal and relevant metrics data and reduced the computation time. Thus the feature selection is core step for any classifier to improve the overall prediction output.

Keywords – *defect prediction model, software quality, feature selection, software metrics, machine learning algorithms.*

I. INTRODUCTION

The Software defect prediction has gained the lot of attention in the field of software engineering and research. It is developed using the software metrics (attributes) which depends on the data collected from the previous software projects. The effective prediction models were constructed using pre-processing of data which includes the task of data cleaning, feature selection, clustering, analysis of redundancy etc. The cleaned datasets are then used as input independent variables for training the classifiers. It uses limited resources to predict the defects prone code with software metrics. Software metrics or features show the properties of software modules [1]. Some metrics may be more close to the class (faulty or non-faulty) than others and some may be repeated or not relevant. Hence, the feature selection methods are used to train the models .The need to apply the feature selection is given below [2]:

i. Reduced size of data applied to searching techniques
ii. Redundant and unrelated metrics produce less accurate and complicated models.
iii. Reduce the computation time of model

It has basically two goals - to enhance the accuracy of classifier and lessen the features count. These are mutually incompatible with each other. The prediction models were developed using many machine learning algorithms [3] such as Decision Tree, Naïve Bayes, KNN, Neural Network, SVM and Random Forest .They have

March - April 2020
ISSN: 0193-4120 Page No. 2245 - 2253

used different methods for selecting the suitable subset of metrics with respect to their context and the outcomes vary according to scenario.

Many comparative studies were performed using feature selection methods in the literature. For example, earlier study conducted with nine different classifiers and artificial immune systems using five NASA datasets with 94 class level metrics and 21 method level metrics for each dataset. The capability of correlation based feature selection technique was evaluated [4]. Similarly previous empirical study conducted in three scenarios using six prediction models (J48, LR, NB, DT, SVM and BN),the ability of correlation based feature selection and Greedy Stepwise algorithm was validated .ANOVA test were conducted for consistency among the predictors on 34 datasets [5]. Earlier experimental study conducted the two-stage data pre-processing approach on three different classifiers (NB, IB1, C4.5) using the six different schemes of filter based feature selection with respect to the relevance and similarity measures. The power of ranking methods was validated using the Friedman test for consistency [6]. Previous empirical study [7] revealed potential of 32 different feature selection methods conducted on NASA datasets using Random Forest classifier which has good prediction capability. These methods were evaluated using the Friedman statistical test and Scott-Knott multiple comparison test . For instance , before cleaning CM1 release have 40 features, 505 modules,9.50 % defect proneness and after cleaning it have 37 features ,327 modules and 12.84 % defect proneness. This demonstrated the impression of feature selection on prediction.Again the ability of clustering-based feature method with the information gain was evaluated in [8] by using deep fuzzy concept . For instance , Eclipse 2.0 have original 155 features and using RUS only 81 features selected for learning the classifiers .Another study compared and revealed that multi-objective feature selection outperformed with 22 filter-based and wrapper-based methods in terms of fewer metrics selection and computation time [9]. Earlier study [10] highlighted the issue of manual selected features and adopted the just-in –time method which is based on neural forest. They combined two heterogeneous methods and performed five scenarios to compare the proposed method. Performance validated using precision, AUC, recall,F1-score and P-opt and revealed the high valued feature representation than the previous methods using open source projects datasets.

The purpose of this literature review is to identify and analyse various feature selection techniques used by predictors in 24 studies conducted between 2009 and 2019. Our review investigates how the selected subset of metrics affected the performance of models, which feature selection methods were used in the different scenarios for choosing the relevant subset of metrics, the independent variables of metrics and dependant variable used, the training datasets, performance measures used among different classifiers. This review shows the comparative study of all these aspects in our work. It helps future studies to make decisions on choosing the suitable methods of features selection in their scenario.

The remaining paper is organized as follows: Section 2 described the Feature Selection with its methodologies and related literature. Section 3 presented the discussion based on the literature. In section 4, conclusion and future directions are presented.

II. FEATURE SELECTION

Feature Selection and data cleaning should be the first and most important step for any SDP model. The purpose of feature selection is: 1) To improve performance of prediction, 2) To reduce computation time and cost-saving and 3) To under the underlying process that generated the data. The benefits of feature selection are to facilitate visualization of data, to reduce storage space requirement, to reduce learning and utilization times and dimensionality curse.

Features are also independent variables or predictor variables input to predictors that will causes the one dependent variable in machine learning and statistics .The dependent variable is to the response variable to find the defect-prone label as faulty or non-faulty [5]. Its task is to reduce irrelevant ,redundant and noisy features .The irrelevant features provides no useful information and redundant features takes more time for computation and provides no more information than the selected unique features [11].

A. *Feature Selection Methods*

The broad classification of feature selection techniques includes filter, wrapper, embedded, hybrid approaches [11] along with evolutionary search techniques [24], ensemble learning approach[28][32] as shown in figure 1 are used in our literature. These are incorporated into many studies depending on different context. This literature describes some previous methods considered from 2009 to 2019 for feature selection. For analysis, the title and abstract of the paper with keywords were collected for initial search. The aspects for selection of papers like to improve the defect detection rate, feature selection methods used in different context, independent variables of code metrics and one dependent variable , training datasets , performance measures used.

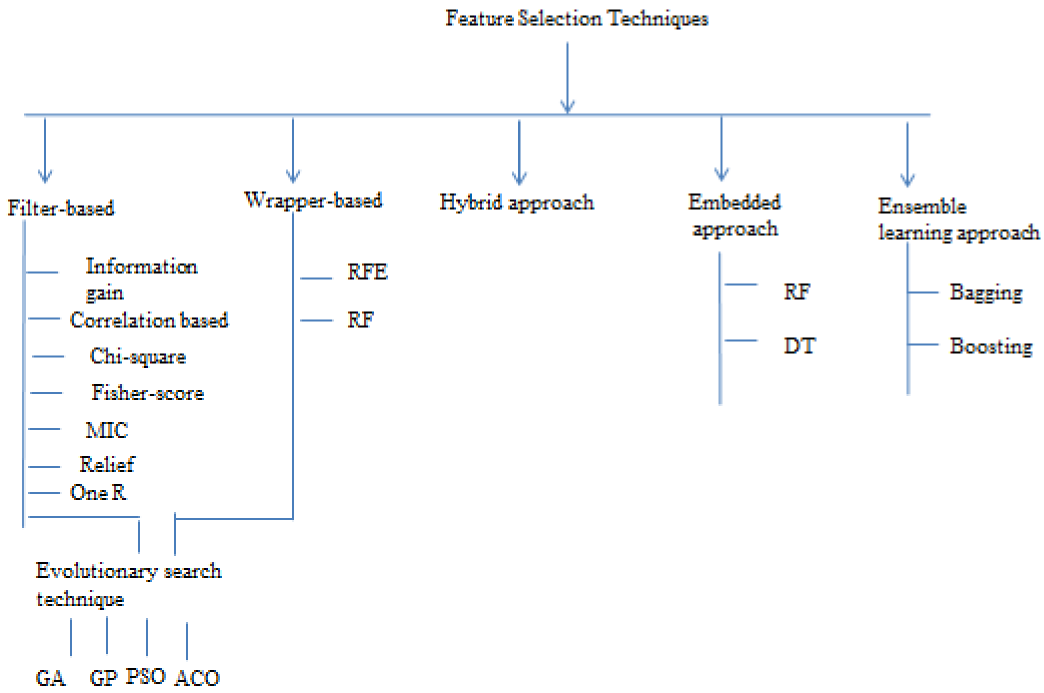

Figure 1. Feature Selection Techniques

B. *Based on Filter-based models*

This approach does not use any learning algorithm for analysis of characteristics of data and evaluation of features .It contains independent measures to evaluate the features subset. This approach is fast and efficient for computation .The methods used for this approach are – information gain , chi-square , fisher score, correlation based, ReliefF, OneR ,maximal info co-efficient .

Previous study [4] conducted with nine different classifiers and artificial immune systems using five NASA datasets-CM1,JM1,KC1,KC2,PC1

March - April 2020
ISSN: 0193-4120 Page No. 2245 - 2253

with 94 class level metrics and 21 method level metrics for each dataset. They used correlation based feature selection technique to choose the suitable subset of metrics . Out of 21 metrics 7 for CM1 ,8 for JM1,8 for KC1,3-KC2, 6-PC1 were selected. Also the seven test groups were conducted on different classifiers. The AUC evaluated the random forests outperformed for large datasets and Naïve Bayes for small datasets .Similarly, in [5] the three scenarios using six prediction models (J48, LR, NB, DT, SVM and BN) evaluated the results of three types of predictors using the twenty different metrics as independent variables and one dependant variable. They used CfsSubsetEval feature selection and Greedy Stepwise algorithm with one-way ANOVA test to check the consistency among the classifiers. They collected 34 datasets from PROMISE repository such as Ant 1.3 release have 125 instances ,20 defects and 16% defects proneness. The Naïve Bayes classifier has the good performance as compared to other classifiers and the small subset of metric improved the accuracy of the classifiers in different scenarios. According to[6] they conducted the two-stage data pre-processing approach on three different classification models(NB, IB1, C4.5) using six different schemes for feature selection with respect to the relevance and similarity measures .Out of these IG outperformed the other ranking techniques and top k features selected. CFS selected the highly relevant metrics. The random under sampling methods used to reduce the instances and make balanced dataset. This study was carried out on three Eclipse datasets each containing 155 features,25,210 instances ,3397 fault prone instances and ten NASA datasets containing 401 features ,23,010 instances , 1225 fault prone instances. The most significant 10 features were selected and compared the performance of classifiers with other classifiers using AUC measures. The 89 % of features and 60 % of instances were reduced to simplify the learning process of classifiers. In [13] used threshold-based feature selection technique with mutual information, deviance, PRC, AUC and Kolmogorov- Smirnov. The classifier used was logistic regression, k-NN, Multilayer Perceptron. ANOVA tests stated it performed best and resulted into the best features above 98.5 % features reduced and compared to no feature selection on Eclipse datasets. In [14] they employed maximal information co-efficient and the grouped features with hierarchical agglomerative clustering. This method is compared with the three classic methods such as GR, chi-square and ReliefF. This method showed the better results as AUC with 0.780 for naïve bayes , f-measure with 0.50 for random forest and RIPPER with 0. 669 on NASA and AEEEM datasets. In [15] they incorporated the information gain feature selection with rough set-based k-NN rule for noise filter before executing simple ensemble rough k-NN. Here Eclipse and NASA software projects datasets were used for experimentation .The comparison performed using Friedman test and Wilcoxon test for evaluation of the proposed method. The statistical analysis demonstrated this study helped for enhancing the performance of prediction models. In [16] the feature selection method is combined with Resample and SMOTE, analysed the performance of various classifiers using the sampling techniques- resample and SMOTE. The fisher linear discriminant analysis used to select the most effective features. The comparison is made between Resample and SMOTE using fisher linear analysis and gave better performance in term of recall, precision AUC and f-measure. The study is carried out using 15 publicly available datasets. From the above literature it is observed that correlation-based, information gain, chi-square ,Relief methods have gained more attention in the filter-based approach.

C. *Based on Wrapper-based models*

This approach uses predetermined learning algorithm and used it to evaluate the identified relevant features .The different algorithms can be produced by generating the subset and evaluated using dependent criterion. It selects an optimal subset of features and hence its performance is better but computation is expensive. The methods used for this approach are –recursive feature elimination, genetic algorithms.

According to [17] wrapper-based attribute ranking methods and RUS on the majority class used to reduce the negative effect of imbalanced data on the prediction models. The eight datasets were used, four from telecommunication software systems (TC1,TC2,TC3,TC4) and four from NASA software projects(CM1,JM1,MW1 and PC1).This study revealed that for imbalanced datasets ,attribute selection becomes more efficient when used after data sampling. ANOVA test was conducted in five different context to evaluate the performance of ranking methods. Similarly in [18] they used decision rule induction method and performance of the 18 classifiers measured with this technique. The comparison done with SVM and RELIEF method and effectiveness of method checked with the help of ROC ,RMSE , MAE. The dataset used is the class level dataset named KC1 which contains class level metrics and method level metrics. The result shows that out of 94 features only 15 features were selected using this proposed method. According to [19] used three different wrapper based algorithms such as binary ant colony optimization, binary genetic algorithm and binary particle swarm optimization to perform with layered recurrent neural network classifiers .The performance tested with Wilcoxon statistical test and outperformed with other classifiers using the PROMISE software repository datasets .In [20] used the optimization with adaptive synthetic sampling approach. The transfer function used to convert the moth flame optimization from continuous to binary manner. The selected features by EBMFOV3 of ant 1.7 dataset release from 20 to 5 ,having 0.76 AUC value as best. The highest AUC is 67% for one algorithm and other algorithm produced less number of selected features in 87 % of datasets.

D. *Based on Hybrid approach*

The additional hybrid approach is proposed which is the combination of filter and wrapper methods. It is also called as greedy randomized adaptive search procedure (GRASP). It revealed the multidisciplinary problem nature .It overcome the drawbacks of one approach by using the advantage of the other one. The plus point is that it derived the positive approach of both models to improve the performance of feature selection. This method used in[21] where the first step is to filter the feature sets and the resulted features as the input wrapper feature selection. This method is compared with simple filter and wrapper methods for evaluation purpose and outperforms than both. The KC1 NASA's dataset has a total of 2109 modules with 326 defective instances and 22 attributes. Result displayed that naïve bayes has good prediction of 85.59%.Previous study [22] proposed the two-stages for hybrid selection method in which first stage groups the features with the help of clustering in the form of hierarchical agglomerative and second stage selects the dominating features from each cluster by removing unwanted features. This paper extends MICHAC [14] and better performed than MICHAC and used two wrapper based techniques. The NASA datasets were used for evaluation and performance metrics used were AUC, recall, precision and F-measure. The proposed system compared with six filter-based feature selection methods and two classic wrapper feature selection methods and HFS demonstrated higher performance using AUC and F-measure. Similarly another study[31] used the non-linear manifold detection method for choosing and minimizing the metrics performed using Random Forest and Decision tree. This study carried out using four datasets having 21 features each. Camel dataset having 19.48 % defective modules ,after applying this method on decision tree

March - April 2020
ISSN: 0193-4120 Page No. 2245 - 2253

classifier showed highest accuracy of 80.51%.The results were validated by Freidman and Wilcoxon test and proved it achieved better accuracy as compared to other methods.

E. Based on Embedded models

This approach is somewhat similar to wrapper methods; it is also used to optimize the objective function. It uses the intrinsic metric during learning. The methods used for this approach are decision tree, L1 regularization and random forest. In [23] they applied the Markov blanket principle based on embedded feature selection, that is natural extension to BN theory .The purpose of Markov blanket is to reduce the set of available features and used the HITON algorithm .Here comparison done with and without MB .15 and MB0.5 using Bonferroni-Dunn test and Friedmann test used. Also Augmented Naive Bayes classifiers compared with Naive Bayes classifier using NASA and Eclipse datasets.

Also some studies were carried out using Evolutionary search techniques in different context. As feature selection is considered as multi-objective problem, evolutionary techniques has greatly contributed to it. It has methods such as genetic programming, genetic algorithms, ant colony optimization and particle swarm optimization [24].For example, in [25] they combined particle swarm optimization with bagging technique for improving the accuracy of model. The comparison performed between with and without PSO - Bagging method shown this approach performed well and used t-Test for evaluation of 11 classifiers using NASA datasets .According to [26] used BA algorithm with correlation based feature selection for evaluation of subset of metrics and random forest for prediction purpose. The tera-PROMISE repository were used ,comparison made between BA and Ant search algorithm and BA performed better than ant algorithm .In [27] they employed the fitness function (linear classifier) with the help of genetic evolutionary learning for feature selection. This study is carried out with the Random forest algorithm for classification to find the training data fitness using Bugzilla dataset. The result showed the best fitness as 4371720.18 to generation 3499.

Some studies were carried out using Ensemble learning for feature selection where various feature selection methods merged to produce powerful output. It involves methods such as random forest and support vector machines. [28].The ensembles learning is divided into two ways : homogenous, where similar type of feature selection methods used different training subset of data and heterogeneous where dissimilar feature selection methods used over alike training set of data[32]. For example, in [29] they combined correlation- based selection methods with other approaches. Its performance compared with conventional methods such as weighted SVMs and random forests on the datasets such as Ant-1.7, KC3, MC1, PC2, PC4, and Camel-1.6. Also in [30] three layers of ensemble learning formed by combining various methods together in three stages and achieved robust output RaF - 0.0986 , BaG - 0.981 evaluated in AUC metrics.

III. DISCUSSION

The following points are discussed from the above mentioned literature review.

1. Most of the studies on feature selection methods were carried out recently with the improvements in previous work done to enhance the performance of the software defect prediction.
2. The filter-based feature selection methods mostly used around 60% in many studies. Some studies have used wrapper-based methods were computation cost is not an issue. Few studies used genetic algorithms and ensemble methods .The comparative study of the literature review is shown in table 1.

March - April 2020
ISSN: 0193-4120 Page No. 2245 - 2253

3. Automated techniques required for feature extraction and selection of relevant features which have impact on performance of the classifiers.
4. The various performance measures were used for evaluation of the prediction models such as ROC, recall, accuracy, arithmetic mean, PRC, precision and widely used is AUC measures for prediction. The ANOVA test is widely used to validate the feature selection methods.

TABLE I. COMPARATIVE STUDY

Sr. no	Feature selection Approach	Literature reference	Examples
1	Filter-based	[4],[5],[6],[7],[8],[9],[10],[11], [13],[14],[15],[16]	IG, Correlation-based, gain ratio ,chi-square, MIC, OneR, Relief
2	Wrapper-based	[7],[9],[11],[17], [18],[19],[20]	C4.5,RFE
3	Hybrid approach	[11],[21],[22],[31]	Combined filter and wrapper method
4	Embedded approach	[23]	Decision Tree, Random Forest
5	Evolutionary search approach	[24],[25],[26],[27]	Genetic programming ,Genetic algorithms, ant colony optimization, particle swarm optimization
6	Ensemble learning approach	[29],[30],[32]	Random Forest, SVM, ,bagging, boosting

IV. CONCLUSION

The software quality can be achieved using Software Defect Prediction and plays a vital role in the Software development life cycle .The feature selection is a core part of SDP to improve the overall performance of the classifiers to detect the faulty and non-faulty module. From this literature review ,we have studied the work done from 2009 to 2019 in the feature selection methods. It performed well and contributed to allocate the suitable subset of metrics data and reduced the execution time. Thus the feature selection is core step for any classifier to improve the overall performance of predictors. The filter-based methods contributed more than the other methods in the literature study. In the future, we would prefer to present the generalized and automated methods for extraction and selection of the effective features for software defect prediction.

V. REFERENCES

1. K. Gao, T. M. Khoshgoftaar, H. Wang and N. Seliya "Choosing software metrics for defect prediction: an investigation on feature selection techniques" Software – Practice And Experience Softw. Pract. Exper. 2011; 41:579–606
2. D. Rodríguez, R. Ruiz, J. Cuadrado-Gallego, and J. Aguilar-Ruiz, "Detecting fault modules applying feature selection to classifiers," *2007 IEEE Int. Conf. Inf. Reuse Integr. IEEE IRI-2007*, pp. 667–672, 2007, doi: 10.1109/IRI.2007.4296696.
3. N. Kalaivani and R. Beena, "Overview of Software Defect Prediction using Machine Learning Algorithms," *Int. J. Pure Appl. Math.*, vol. 118, no. 20, pp. 3863–3873, 2018.
4. C. Catal and B. Diri, "Investigating the effect of dataset size, metrics sets, and feature selection techniques on software fault prediction problem," *Inf. Sci. (Ny).*, vol. 179, no. 8, pp. 1040–1058, 2009, doi: 10.1016/j.ins.2008.12.001.
5. P. He, B. Li, X. Liu, J. Chen, and Y. Ma, "An empirical study on software defect prediction with a simplified metric set," *Inf. Softw.*

Technol., vol. 59, pp. 170–190, 2015, doi: 10.1016/j.infsof.2014.11.006.

6. W. Liu, S. Liu, Q. Gu, J. Chen, X. Chen, and D. Chen, “Empirical Studies of a Two-Stage Data Preprocessing Approach for Software Fault Prediction,” *IEEE Trans. Reliab.*, vol. 65, no. 1, pp. 38–53, 2016, doi: 10.1109/TR.2015.2461676.
7. Z. Xu, J. Liu, Z. Yang, G. An, and X. Jia, “The Impact of Feature Selection on Defect Prediction Performance: An Empirical Comparison,” *Proc. - Int. Symp. Softw. Reliab. Eng. ISSRE*, no. April 2018, pp. 309–320, 2016, doi: 10.1109/ISSRE.2016.13.
8. A. Arshad, S. Riaz, L. Jiao, and A. Murthy, “The empirical study of semi-supervised deep fuzzy c-mean clustering for software fault prediction,” *IEEE Access*, vol. 6, pp. 47047–47061, 2018, doi: 10.1109/ACCESS.2018.2866082.
9. C. Ni, X. Chen, F. Wu, Y. Shen, and Q. Gu, “An empirical study on pareto based multi-objective feature selection for software defect prediction,” *J. Syst. Softw.*, vol. 152, pp. 215–238, 2019, doi: 10.1016/j.jss.2019.03.012.
10. Y. Qiu, Y. Liu, A. Liu, J. Zhu, and J. Xu, “Automatic Feature Exploration and an Application in Defect Prediction,” *IEEE Access*, vol. 7, pp. 112097–112112, 2019, doi: 10.1109/access.2019.2934530.
11. H. Liu and L. Yu, “Toward Integrating Feature Selection Algorithms for Classification and Clustering,” *Knowl. Creat. Diffus. Util.*, vol. 17, no. 4, pp. 491–502, 2005, doi: 10.1109/TKDE.2005.66.
12. H. (National U. of S. Liu, H. (Osaka U. Motoda, R. Setiono, and Z. Zhao, “Feature Selection : An Ever Evolving Frontier in Data Mining,” *J. Mach. Learn. Res. Work. Conf. Proc. 10 Fourth Work. Featur. Sel. Data Min.*, pp. 4–13, 2010.
13. H. Wang, T. M. Khoshgoftaar, and N. Seliya, “How many software metrics should be selected for defect prediction?,” *Proc. 24th Int. Florida Artif. Intell. Res. Soc. FLAIRS - 24*, no. Mi, pp. 69–74, 2011.
14. Z. Xu, J. Xuan, J. Liu, and X. Cui, “MICHAC: Defect prediction via feature selection based on Maximal Information Coefficient with Hierarchical Agglomerative Clustering,” *2016 IEEE 23rd Int. Conf. Softw. Anal. Evol. Reengineering, SANER 2016*, vol. 2016-Janua, pp. 370–381, 2016, doi: 10.1109/SANER.2016.34.
15. S. Riaz, A. Arshad, and L. Jiao, “Rough Noise-Filtered Easy Ensemble for Software Fault Prediction,” *IEEE Access*, vol. 6, pp. 46886–46899, 2018, doi: 10.1109/ACCESS.2018.2865383
16. A. Kalsoom, M. Maqsood, M. A. Ghazanfar, F. Aadil, and S. Rho, *A dimensionality reduction-based efficient software fault prediction using Fisher linear discriminant analysis (FLDA)*, vol. 74, no. 9. Springer US, 2018.
17. T. M. Khoshgoftaar and K. Gao, “Feature selection with imbalanced data for software defect prediction,” *8th Int. Conf. Mach. Learn. Appl. ICMLA 2009*, no. May 2014, pp. 235–240, 2009, doi: 10.1109/ICMLA.2009.18.
18. N. Gayatri, S. Nickolas, and A. V Reddy, “Feature Selection Using Decision Tree Induction in Class level Metrics Dataset for Software Defect Predictions,” *Lect. Notes Eng. Comput. Sci.*, vol. 2186, no. 1, pp. 124–129, 2010.
19. H. Turabieh, M. Mafarja, and X. Li, “Iterated feature selection algorithms with layered recurrent neural network for software fault prediction,” *Expert Syst. Appl.*, vol. 122, no. January 2019, pp. 27–42, 2019, doi: 10.1016/j.eswa.2018.12.033.
20. I. Tumar, Y. Hassouneh, H. Turabieh, and T. Thaher, “Enhanced Binary Moth Flame Optimization as a Feature Selection Algorithm to Predict Software Fault Prediction,” *IEEE Access*, pp. 1–1, 2020, doi: 10.1109/access.2020.2964321.
21. C.A. Devi, K.E. Kannammal and B .Surendiran “A Hybrid Feature Selection Model For Software Defect Prediction “ IJCSA Vol 2 No.2, 2012
22. Y. Jian, X. Yu, Z. Xu, and Z. Ma, “A hybrid feature selection method for software fault prediction,” *IEICE Trans. Inf. Syst.*, vol. E102D, no. 10, pp. 1966–1975, 2019, doi: 10.1587/transinf.2019EDP7033.
23. \K. Dejaeger, T. Verbraken, and B. Baesens, “Toward comprehensible software fault prediction models using bayesian network classifiers,” *IEEE Trans. Softw. Eng.*, vol. 39, no. 2, pp. 237–257, 2013, doi: 10.1109/TSE.2012.20.
24. B. Xue, M. Zhang, W. N. Browne, and X. Yao, “A Survey on Evolutionary Computation Approaches to Feature Selection,” *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 606–626, 2016, doi: 10.1109/TEVC.2015.2504420.

March - April 2020
ISSN: 0193-4120 Page No. 2245 - 2253

25. R. S. Wahono and N. Suryana, "Combining particle swarm optimization based feature selection and bagging technique for software defect prediction," *Int. J. Softw. Eng. its Appl.*, vol. 7, no. 5, pp. 153–166, 2013, doi: 10.14257/ijseia.2013.7.5.16.
26. D. R. Ibrahim, R. Ghnemat, and A. Hudaib, "Software defect prediction using feature selection and random forest algorithm," *Proc. - 2017 Int. Conf. New Trends Comput. Sci. ICTCS 2017*, vol. 2018-Janua, no. October, pp. 252–257, 2017, doi: 10.1109/ICTCS.2017.39.
27. P. Patchaiammal and R. Thirumalaiselvi, "Genetic evolutionary learning fitness function (GELFF) for feature diagnosis to software fault prediction," *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 11 Special Issue, pp. 1151–1161, 2019, doi: 10.35940/ijitee.K1233.09811S19.
28. Y. Saeys, T. Abeel, and Y. V. de Peer "Robust Feature Selection Using Ensemble Feature Selection Technique " Springer ECML PKDD 2008, Part II, LNAI 5212, pp. 313–325
29. I. H. Laradji, M. Alshayeb, and L. Ghouti, "Software defect prediction using ensemble learning on selected features," *Inf. Softw. Technol.*, vol. 58, no. September 2019, pp. 388–402, 2015, doi: 10.1016/j.infsof.2014.07.005.
30. C. W. Yohannese, T. Li, and K. Bashir, "A three-stage based ensemble learning for improved software fault prediction: An empirical comparative study," *Int. J. Comput. Intell. Syst.*, vol. 11, no. 1, pp. 1229–1247, 2018, doi: 10.2991/ijcis.11.1.92.
31. S. Ghosh, A. Rana, and V. Kansal, "A Hybrid Nonlinear Manifold Detection Approach for Software Defect Prediction," *2018 7th Int. Conf. Reliab. Infocom Technol. Optim. Trends Futur. Dir. ICRITO 2018*, pp. 453–459, 2018, doi: 10.1109/ICRITO.2018.8748788.
32. V.B-Canedo,A.A-Betanzos "Ensembles for feature selection:A review and future trends" Information Fusion 52(2019)1-12 ELSEVIER 1566-2535