Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CROSS-PROJECT SOFTWARE DEFECT PREDICTION

YAHAYA ZAKARIYAU BALA[1], PATHIAH ABDUL SAMAT[2], KHAIRONI YATIM SHARIF[3], NORIDAYU MANSHOR[4]

[1]PhD. student, Software Engineering, Universiti Putra Malaysia, Malaysia
[2]Senior Lecturer, Software Engineering, Universiti Putra Malaysia, Malaysia
[3]Senior Lecturer, Software Engineering, Universiti Putra Malaysia, Malaysia
[4]Senior Lecturer, Computer Science, Universiti Putra Malaysia, Malaysia
[1]Lecturer I, Computer Science, Federal University of Kashere, Nigeria
E-mail: [1]balagombi2@gmail.com, [2]pathiah@upm.edu.my, [3]khaironi@upm.edu.my, [4]ayu@upm.edu.my

ABSTRACT

The feasibility of building a software defect prediction (SDP) model in the absence of previous records has been increased by the introduction of the Cross-Project Defect Prediction (CPDP) method. Although this method overcomes the limitations of SDP in the absence of previous historical records, the predictive performance of the CPDP model is relatively poor due to distribution discrepancy between the source and the target datasets. To overcome this challenge, various studies have been published. This SLR was conducted after analyzing research articles published since 2013 in four digital libraries: Scopus, IEEE, Science Direct, and Google Scholar. In this work, five research questions covering the classification algorithms, dataset, independent variables, performance evaluation metrics used in CPDP studies, and as well as the performance of individual machine learning classification algorithms in predicting software defects across different software projects were addressed accordingly. To respond to outlined questions, 34 most relevant articles were selected after passing through quality assessment criteria. Through this work, it was discovered the majority of the selected studies used machine learning techniques as classification algorithms, and 64% of the studies used the combination of Object-Oriented (OO) and Line of Code (LOC) metrics. All the selected studies used publicly available datasets from NASA, PROMISE, SOFLAB, AEEEM, and Relink. The most commonly used evaluation metrics are F_measure and AUC. Best performing classifiers include Logistic Regression and SVM. Despite various efforts to improve the performance of the CPDP model, the performance is below the applicable level. Thus, there is a need for further study that will improve the performance of the CPDP model.

Keywords: *Software Defect Prediction, Cross-Project, Machine Learning Techniques, Statistical Techniques, Performance Evaluation Measure.*

1. INTRODUCTION

Software Defect Prediction (SDP) is one of the hot topics in software engineering that deals with the process of identifying defective components of software projects. SDP reduces the testing effort by directing the test team toward the faulty part of the software instead of visiting every part of the software. Also, it reduces the cost of maintenance. SDP has become one of the attractive research topics in software engineering [1]. It has drawn the attention of many researchers from both industry and academic settings. Recently, numerous approaches on software defect prediction have been proposed by different researchers using various machine learning classification techniques [2], [3], [4], [5], [6], [7], [8], [9], [10], etc. The process of software defect prediction involves two stages; data pre-processing and model construction (defining the relationship between the dependent and independent variable i.e., defects and metrics respectively). An SDP performed both training and testing (source and target) on the same historical data obtained from the part of the same project termed Within Project Defect Prediction (WPDP) [11]. The main problem in such models is the lack of enough historical data, especially for new projects. In order to alleviate the problem observed in WPDP, different researchers proposed CPDP. [11] proposed the earliest study found in the literature in which a model trained on historical data of one java project(source) and predicted defects in another java project (target). Machine learning and statistical techniques have been used to construct the CPDP models in different studies. To have an insight into what has been done so far on CPDP, it is important to make a summary

Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

of the existing literature. Therefore, this systematic literature review has analyzed and summarized the classification techniques used, independent variable (metrics), datasets used, and evaluation measures used by reviewing the studies published between 2013 and 2021 on CPDP. Hosseini *et al.* [1] conducted SLR on CPDP to know the state-of-the-art in CPDP and the corresponding metrics, datasets, and models. This work is more concrete than the existing one in the sense that the performances of different classification techniques have been empirically analyzed in a more comprehensive way using the F-measure which appeared to be the most commonly used performance evaluation measure in CPDP studies. For the purpose of achieving the aim of this work, 34 primary studies were considered after passing the strict quality assessment.

The paper is structured as follows: Sect. 2 described the planning stage includes; identifying reasons for the review, forming research questions, devising a suitable method for searching for relevant studies, designing inclusion and exclusion criteria, and forming criteria for quality assessment. Sect. 3 describes the selected studies with the quality analysis The yearly distribution of the studies and Journal/Conference distribution are presented. Sect. 4 presents how the research questions were answered in this paper. Sect. 5 present the limitations of the review. Sect. 6 Reports the conclusions and future work.

1.1 Prior Work

In 2016, Hosseini et al [14] performed a systematic review of cross-project defect prediction with the aim to synthesize existing literatures published up till 2015 to understand the state-of-the-art in CPDP in terms of metrics, models, data approaches, datasets and associated performances, but many current publications on CPDP are not part of their review. For instance, in this work, only six (6) publication out of thirty-four (34) selected articles were published before 2016. Thus, there is need for SLR to track research trends on CPDP.

2. METHODOLOGY

2.1 Review Method

The guidelines proposed by [12] have been strictly followed in this review. As can be seen in Figure 1 below, the review was done by following the process. As stated in [12], a systematic review constitutes three steps; planning, conducting, and reporting. The planning stage includes; identifying reasons for the review, forming research questions, devising a suitable method for searching relevant studies, designing inclusion and exclusion criteria, forming criteria for quality assessment designing form for extraction and synthesis of information method for performing the review and evaluation as well. The next step is performing the review where the search method, extraction, and data synthesis are implemented. Search method, inclusion and exclusion criteria, data extraction, and synthesis method in detail.

2.2. Research Questions

The reason for this work is to systematically review the CPDP studies with respect to software metrics, classification models, evaluation methods, and datasets used. Table 1 below contains three research questions. In RQ1 the classification techniques that are being used for constructing CPDP models have been synthesized. RQ2 determines metrics used in CPDP. RQ3 finds the datasets used in CPDP. RQ4 finds the performance evaluation measure used in CPDP. RQ5 determines how the various classification techniques used for CPDP models perform.

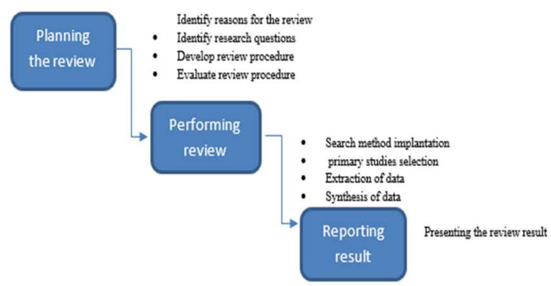

Fig. 1: *Systematics Literature review process*

2.3. The Strategy of Searching and Selection of the Study

The aim of the search strategy is to find all relevant primary studies that will be able to answer research questions. This begins by defining search strings, selecting databases, and finally stating the inclusion and exclusion criteria. Alternative spellings, synonyms, and Boolean ANDs and ORs were considered for constructing search terms. The following search term was used: (“cross-project” OR “cross-project” OR “multiple-project” OR “multiple projects” OR “mixed-project” OR “mixed project”) AND (bug* OR fault* OR defect* OR error*) AND (prediction* OR estimation*) AND software. Below are the databases used for searching primary studies;

- IEEExplore
- Science Direct
- Scopus
- Google Scholar

Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The search was done on 6th May 2021, and studies identified have been published from 2012 up to date.in Science Direct, the Boolean "OR" and "AND" were reduced to a maximum of eight(8) and the wildcard "*" was also removed before the string was accepted as shown below; ("cross-project" OR "multi-project") AND (defect OR fault OR bug) AND (predict OR estimate) AND software In Google Scholar, thought the search string was accepted but the output was not relevant therefore it was reduced, as shown below;

("cross-project" OR "cross-project" AND (defect OR fault OR bug) AND (predict) AND software. A total of 485 articles were obtained (Scopus: 166, ScienceDirect:99 and Google Scholar; 270) before applying the criteria. 37 studies were considered after going through the abstract and introduction of the studies, the inclusion-exclusion criteria were then applied to the selected studies.

2.4. Data Extraction

Empirical studies on CPDP (learning and prediction, dataset, and evaluation techniques) are to be included. Predicted results must contain information related to defects, such as defect labels. Furthermore, papers have to be written in the English language. peer-reviewed journal papers and conference papers on CPDP, in particular, are to be included.

Inclusion Criteria

- Empirical CPDP Studies
- CPDP Studies that have been written in English language

Exclusion criteria

- Reviews studies.
- Papers that are not empirical studies.
- Papers not written in English language.

2.5. Quality Assessment Criteria

The importance of quality of the selected studies cannot be over emphasised. Therefore, in this review, 8 quality questions were designed.

Table 1: Research Questions

ID	Questions	Description	Objective
RQ1	Which classification techniques used for CPDP models?	Techniques used to define the relationship between dependent and independent variable	To know classification models used in CPDP studies
RQ2	Which metrics have been used in CPDP?	metrics that are being used in CPDP	To examine metrics used for CPDP
RQ3	Which data sets have been used for CPDP?	the set of components and their corresponding metrics value and labels	To know the datasets used in CPDP and their sources
RQ4	Which performance evaluation measures have been used for CPDP?	Method used to measure performance	To know the evaluation methods upon which CPDP is justified
RQ5	What is the performance of various classification techniques used for CPDP models?	Techniques used for CPDP models	To know which technique performed well.

2.5. Quality Assessment Criteria

The importance of quality of the selected studies cannot be over emphasised. Therefore, in this review, 8 quality questions were designed.

2.6. Extraction and Synthesis of Data

Data extraction form was designed using excel in order to keep relevant data to answer the research questions. The form contains the name of the papers, year, metrics, dataset and the evaluation techniques used. The results of research questions are presented using tables and charts.

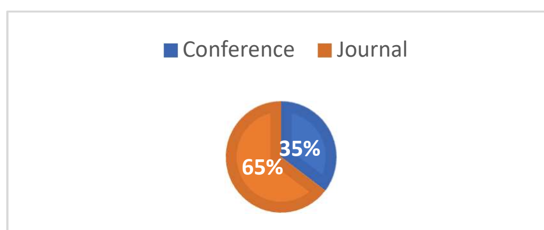

Figure2: *Journal/Conference distribution*

Table 2: Quality Assessment Questions

S/N	Question	Yes	Partially	No
1	Are the study's objectives clearly stated?			
2	Is the study scope clearly stated?			
3	Are there relevant literature in the study?			
4	Are the dependent variable(metrics) clearly stated			
5	Are the independent variable clearly stated?			
6	Are the RQs properly addressed?			

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

7	Has the study been cited?			
8	Are the performance evaluation techniques properly written?			

3. Sources of selected studies and quality analysis

The selected studies with the quality analysis are described. The yearly distribution of the studies and Journal/Conference distribution are presented.

3.1. Explanation of Selected Primary Studies and Quality Analysis

The selected primary studies are described in this section. After passing inclusion and exclusion criteria, 34 studies were selected and put into quality assessment criteria (Table 2) as well. Table 2 is having 8 questions. i. e., 8 scores, one for each question. Those have scored 5 or above were selected and below 5 were rejected. The studies that scored from 5 and above 34 articles were selected. A nomenclature was assigned to each study i.e., PS (Primary Study). About 65% were published in journals while 25% in the proceeding of various conferences as shown in figure 2 below.

3.2. Yearly Distribution of Selected Primary Studies

As shown in Figure3, the yearly distribution of the primary studies from 2013 to 2021 study. It can be observed most of the selected studies were published between 2018 to 2019.

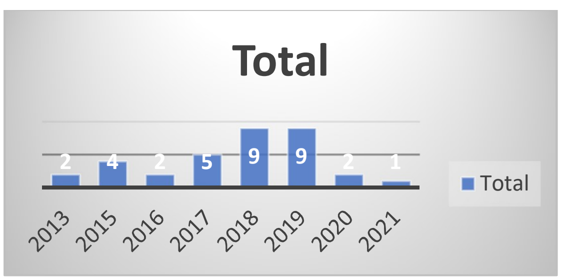

Figure 3: *Publication per year*

4. RESULT

4.1 RQ1.Which Classification Techniques Used for CPDP Models?

To accurately perform the Cross-Project Defect Prediction, numerous techniques have been applied by researchers. These techniques established the relationship between dependent and independent variables to build prediction models. the classification techniques used for CPDP models can be categorized into two:
(i) Machine Learning techniques (ML) (ii) Statistical techniques (ST)

It is obvious from Table 4 that the majority of studies (41%) employed ML techniques for building CPDP models, 29% of the studies applied ST techniques, and 29% of the studies applied the combination of both ML techniques and ST techniques. About 13 primary studies employed multiple techniques to compare the performance of one technique to the other. It was discovered that ML techniques are the commonly used classification techniques for CPDP. Table 5 below presents the distribution of studies that used different ML for construction CPDP. NB was the most frequently used among the primary studies using ML techniques which were around 32% of the studies followed by RF and SVM, which were used in 17% and 15% of the studies, respectively. In the primary studies that used ST techniques for CPDP, 9 studies applied linear regression LR (PS1, PS3, PS4, PS9, PS10, PS5, PS20, PS23, PS33).

4.2. RQ2.Which Metrics Have Been Used In CPDP?

Numerous literature has been using various software metrics to characterize the software system such as complexity, modularity, etc. For CPDP, the ability of software metrics to act as independent variables makes it more applicable. To answer RQ2, the considered primary studies were categorized based on software metrics used as independent variables as given below:
–Selected studies that used Line of Code (LOC) software metrics: PS4, PS16.
–Selected studies using OO software metrics as independent variables: This set of studies used OO metrics as independent variables. In this type of metric, the characteristics of OO systems have been quantified. Selected Studies that used OO metrics are PS29 and PS31.
–Selected studies using Software Change Metrics (SCM) as independent variables; the study is PS11.
–Selected studies using a combination of Object-Oriented software metrics and Line of Code (OO+LOC) as independent variables: the studies are: PS1, PS2, PS3, PS6, PS8, PS10, PS13, PS17, PS18, PS19, PS22, PS23, PS24, PS27, PS32, PS33.
–Selected studies using a combination of Object-Oriented software metrics, Line of Code, and Software Change Metrics (OO+LOC+SCM) as independent variables. the studies are PS12.
–Selected studies using a combination of Line of Code and Cyclomatic Complexity Metrics (LOC+CC) as independent variables. the studies are PS15 and PS26.
–Selected studies using a combination of Entropy, Coupling Between Objects, and Line of Code

Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Metrics (Entropy+CBO+LOC) as independent variables. the study is PS20.
The distribution of studies based on metrics used can be seen in Figure 4. 64% of the studies used OO+LOC metrics, 8% of studies used the combination of LOC+CC metrics, and 8% of studies used OO metrics. Thus, composite of OO and LOC have been found to be the most popularly used metrics for CPDP.

4.3. RQ3. Which Data Sets Have Been Used For CPDP?

A different range of datasets from different sources is used to build models in CPDP. The most widely used datasets are from open sources such as NASA: NASA Metrics Data Program (MDP) comprises 13 datasets at the method of function level software metrics. Metrics in these datasets vary, AEEEM: D'Ambros (2010) extracted the AEEEM suit which contains metric and bug data from five open-source projects. The dataset in this suit contains 61 metrics. Relink: Wu et al. (2011) extracted this suit. Datasets in this suit comprise 26 static code metrics, SoftLab: these datasets were donated by SoftLab and they comprise 29 static code metrics.

*Table **3**: Distribution Of Primary Studies Based On The Classification*

Technique used	**Study Identifier**	**Count**	**Percentage**
ML techniques only	PS2, PS5, PS6, PS8, PS13, PS14, PS16, PS17, PS24, PS25, PS26, PS27, PS29, PS31, PS34	14	41
ST techniques only	PS1, PS3, PS4, PS9, PS10, PS15, PS20, PS23, PS33	10	29
ML techniques & ST techniques	PS7, PS11, PS12, PS18, PS19, PS21, PS22, PS28, PS32	10	29

Table 4: Machine Learning Used In The Selected Studies With Percentage

ML Technique	**No. of Studies**	**Percentage**
NB	13	32
RF	7	17
SVM	6	15
C4.5	3	7
J48	3	7
CNN	2	5
NN	2	5
KNN	1	2
GB	1	2
TrAdaBoost	1	2
GBM	1	2

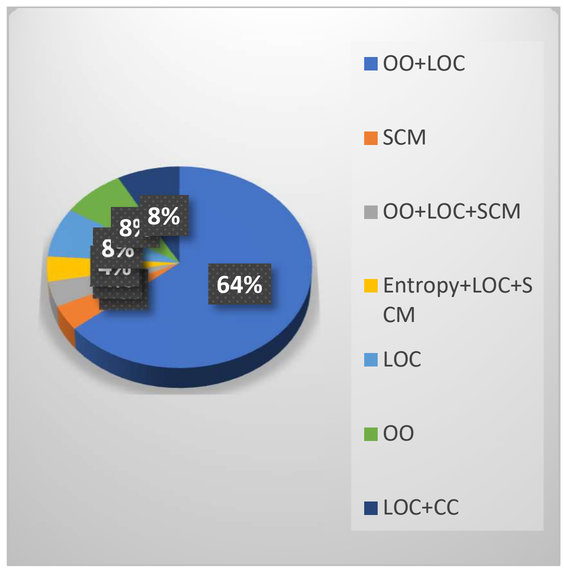

Figure 4: Metrics distribution

4.4. RQ4. Which Performance Evaluation Measures Have Been Used For CPDP?

As shown in Table 6, after analyzing the performance evaluation measure of the 34 selected primary studies, it was discovered that the majority of them used F-measure to evaluate their prediction models. few had adopted the combination of both F-measure and AUC (PS15, PS18, PS25, PS28). Few studies considered AUC only (PS9, PS23, PS32). The reason for selecting a particular evaluation technique varies across studies. Some were based on popularity while some were based on theoretical reasons.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.5. RQ5.What Is the Performance of Various Classification Techniques Used for CPDP models?

To evaluate how the classification techniques discussed in section IV(A) above performed, the F-measure values of different classification techniques on datasets obtained from NASA, AEEEM, PROMISE, SOFTLAB, and Relink were recorded. In order to generalize the result, classification techniques considered were used in at least two primary studies. F-measure was adopted simply because it was the most commonly used evaluation measure in the selected primary studies. The average was computed in a study where more than one dataset was used for the experiment. A statistical measure such as mean, minimum, maximum, and standard deviation of performance was presented as shown in Table 6. As observed in Table 5, Logistic Regression (LR) reported the highest mean F-measure of 0.46, followed by SVM, C4.5, and NB with mean of F-measure of 0.42, 0.39, and 0.38 respectively. Therefore, despite the fact that the mean result of LR in terms of F-measure is below the benchmark of 7.5 as stated by [13] yet LR performed better than the rest of the classification techniques in the selected studies. KNN and CNN with a mean of F-measure 0.32 and 0.32 respectively, performed worst among the techniques.

5. LIMITATIONS OF REVIEW

One of the aims of this systematic review is to evaluate the performance of the CPDP models constructed in the literature. In order to achieve this, digital libraries were utilized and 34 primary studies were considered. Thus, there might be a threat of exclusion of relevant studies. Also, only F-measure was considered for evaluating the performance of CPDP models in this review. Other measures such as AUC, BofP20, Precision, etc. were also used in different studies as evaluation measures. Therefore, there might exist a threat that the performance of the CPDP models may differ using different evaluation measures. The values of the F-measure were extracted from different experimental settings such as datasets and metrics adopted, as a result, there might be a threat that this issue of different settings may affect the performance of CPDP models.

Table 5 Performance Statistics Of CPDP Classification Techniques

Technique	Count	Performance measure	Min	Max	Mean	Std
NB	8	F-measure	0.14	0.54	0.38	0.14
LR	17	F-measure	0.12	0.69	0.46	0.19
RF	8	F-measure	0.11	0.66	0.36	0.18
SVM	5	F-measure	0.11	0.57	0.42	0.23
C4.5	2	F-measure	0.29	0.48	0.39	0.09
J48	3	F-measure	0.26	0.4	0.35	0.07
KNN	3	F-measure	0.32	0.32	0.32	0
CNN	2	F-measure	0.53	0.53	0.32	0

Table 6 Performance Evaluation Measures Used For CPDP In The Selected studies

Study	Evaluation measure	Study	Evaluation measure
PS1	PF, PD, F-measure, AUC, G-measure	PS18	F-measure, AUC
PS2	F-measure, PofB20	PS19	Success rate
PS3	F-measure, Accuracy	PS20	F-measure
PS4	F-measure, G-measure, AUC, EArecall, EAF-measure	PS22	F-measure, NoB20
PS5	F-Measure AUC Cost-Effectiveness	PS23	AUC
PS6	F-measure	PS24	F-measure, AUC, and PofB20
PS7	F-measure	PS25	F-measure, AUC
PS8	F-measure	PS26	PD, PF, G-measure
PS9	AUC	PS27	p-value, AUC
PS10	F-measure, Accuracy	PS28	F-measure, AUC
PS11	F-measure, G-measure, Balance, PD, PF	PS29	Accuracy, Precision, Recall, False Alarm
PS12	Precision, Balance, F-measure, AUC, PD, FPR	PS30	F-measure, AUC, balance
PS13	PD, PF, G-measure, Balance	PS31	Accuracy, Precision, Recall, F-measure
PS14	Recall, AUC, F-measure	PS32	AUC
PS15	F-measure, AUC	PS33	F-measure, precision and recall

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

PS16	F-measure, BoP20	PS34	F-measure, precision and recall
PS17	F-measure, G-measure		

6. CONCLUSION AND FUTURE WORK

In this work, current research trends in CPDP have been tracked through SLR. Papers published since 2013 in four digital libraries such as Scopus, Science Direct, IEEE, and Google Scholar. Five research questions concerning the classification algorithm, independent variables, datasets, performance evaluation metrics, and individual classifiers' performance were outlined and answered accordingly. It was concluded that CPDP addressed the issue of historical records but yet the predictive performance of CPDP models is relatively low. Moreover, the issue of high dimensionality datasets used for CPDP should be dealt with in other to improve the effectiveness of CPDP models.

The open research issues identified in this SLR are as follows;

- It was discovered that the majority of the selected studies used public datasets and open-source projects which is small in size. Therefore, future empirical CPDP studies should focus on using industrial datasets.
– It has been observed from the literature that class imbalance affects the accuracy of the result and only a few studies considered solving class imbalance. Therefore, future empirical studies on CPDP should try to consider the issue of class imbalance.
- It was also discovered that almost all the public datasets are highly dimensional in nature and only a few studies considered feature selection techniques. So future studies on CPDP should try to embrace feature engineering.

FUNDING No funding available

CONFLICT OF INTEREST all authors declare that there is no conflict of interest.

REFERENCE

[1] S. Hosseini, B. Turhan and D. Gunarathna. A systematic literature review and meta-analysis on cross project defect prediction. IEEE Transactions on Software Engineering, Vol. 45, No. 2, 2017, pp. 111-147.

[2] M. Shepperd, D. Bowes, and T. Hall "Researcher bias: The use of machine learning in software defect prediction," IEEE Trans. Softw. Eng., Vol. 42, No. 11, Nov. 2014 pp. 1092–1094.

[3] B. Turhan and A. Bener, "Analysis of naive Bayes assumptions on software fault data: An empirical study," Data Knowl. Eng., Vol. 68, No. 2, 2009, pp. 278– 290,

[4] T. Wang and W. Li. "Naive Bayes software defect prediction model," in Proc. IEEE Int. Conf. Comput. Intell. Softw. Eng., Wuhan, China, 2010, pp. 1–4.

[5] N. Gayatri, S. Nickolas, and A. Reddy. "Feature selection using decision tree induction in class level metrics dataset for software defect predictions," *in Proc. IEEE Int. Conf. Eng. Comput. Sci.*, San Francisco, USA, 2010, pp. 127–149

[6] J. Wang, B. Shen, and Y. Chen. "Compressed C4.5 models for software defect prediction," in *Proc. IEEE Int. Conf. Qual. Softw.*, Xi'an, China, 2012, pp. 13–16.

[7] D. Gray, D. Bowes, N. Davey, Y. Sun, and B. Christianson. "Using the support vector machine as a classification method for software defect prediction with static code metrics," Commun. Comput. Inf. Sci., Vol. 43, No. 2, 2009, pp. 223–234.

[8] Z. Yan, X. Chen, and P. Guo. "Software defect prediction using fuzzy support vector regression," *in Proc. Springer Int. Conf. Symp. Neural Netw.*, Berlin, Germany, Aug. 2010, pp. 17–24.

[9] E. Paikari, M. Richter, and G. Ruhe. "Defect prediction using case-based reasoning: An attribute weighting technique based upon sensitivity analysis in neural networks," Inf. J. Softw. Eng. Knowl. Eng., Vol. 22, No. 6, 2012, pp. 747–768.

[10] M. Thwin and T. Quah. "Application of neural networks for software quality prediction using object-oriented metrics," J. Syst. Softw., Vol. 76, No. 2, 2005, pp. 147–156.

[11] L.C. Briand, W.L. Melo, and J. Wust. "Assessing the applicability of fault-proneness models across object-oriented software projects." IEEE transactions on Software Engineering, Vol. 28, No. 7, 2002, pp. 706-720.

[12] B.A. Kitechenham "Guidelines for performing systematic literature review in software engineering," technical report EBSE-2007- 001, UK, 2007

[13] T. Zimmermann, N. Nagappan, H. Gall, E. Giger, and B. Murphy. "Cross-project defect prediction: a large-scale experiment

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

on data vs. domain vs. process." In *Proceedings of the 7th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on the foundations of software engineering*, August, 2009, pp. 91-100.

[14] S. Hosseini, B. Turhan, and D. "A systematic literature review and meta-analysis on cross project defect prediction." IEEE Transactions on Software Engineering, Vol. 45, No. 2, 2017, pp. 111-147.

Published Studies:

[PS1] Q. Zou, L. Lu, Z. Yang, X. Gu, and S. Qiu. "Joint feature representation learning and progressive distribution matching for cross-project defect prediction. Information and Software Technology, 137, 2021, 106588.

[PS2] C. Liu, D. Yang, X. Xia, M. Yan, X. Zhang. "A two-phase transfer learning model for cross-project defect prediction." Inf Softw Technol Vol. 107, 2019, pp. 125–136.

[PS3] W. Wen, B. Zhang, X. Gu, and X. Ju. "An empirical study on combining source selection and transfer learning for cross-project defect prediction." In: *IBF 2019–2019 IEEE 1st international workshop on intelligent bug fixing*. Institute of Electrical and Electronics Engineers Inc., 2019, pp 29–38.

[PS4] Z. Xu, S. Pang, T. Zhang, X.P. Luo, J. Liu, Y.T. Tang, X. Yu, and L. Xue. "Cross project defect prediction via balanced distribution adaptation-based transfer learning." J Comput Sci Technol Vol. 34, 2019, pp. 1039–1062.

[PS5] S. Ren, W. Zhang, H.S. Munir, and L. Xia. "Dissimilarity Space Based Multi-Source Cross-Project Defect Prediction." Algorithms, Vol. 12, No. 1, 2019, pp. 13.

[PS6] S. Qiu, H. Xu, J. Deng, S. Jiang, and L. Lu. "Transfer convolutional neural network for cross-project defect prediction." Applied Sciences, Vol. 9, No. 13, 2019, pp. 2660.

[PS7] D. Chen, X. Chen, H. Li, J. Xie, and Y. Mu. "Deep learning based cross-project defect prediction." IEEE Access, Vol. 7, 2019, pp. 184832-184848.

[PS8] L. Gong, S. Jiang, L. Bo, L. Jiang, and J. Qian. "A novel class-imbalance learning approach for both within-project and cross-project defect prediction." IEEE Transactions on Reliability, Vol. 69, No. 1, 2019, pp. 40-54.

[PS9] P. He, Y. He, L. Yu, and B. Li. "An improved method for cross project defect prediction by simplifying training data." Math Probl Eng. Vol. 2018, 2018, pp. 1–18.

[PS10] S. Qiu, L. Lu, and S. Jiang. "Multiple-components weights model for cross-project software defect prediction." IET Softw Vol. 12, No. 2018, pp. 345– 355.

[PS11] N. Limsettho, K.E. Bennin, J.W. Keung, H. Hata, and K. Matsumoto. "Cross project defect prediction using class distribution estimation and oversampling." Information and Software Technology, Vol. 100, 2018, pp. 87-102.

[PS12] F. Zhang, A. Mockus, I. Keivanloo, and Y. Zou. "Towards building a universal defect prediction model with rank transformed predictors." Empirical Software Engineering, Vol. 21, No. 5, 2016, pp. 2107-2145.

[PS13] D. Ryu, J.I. Jang, J. Baik. "A transfer cost-sensitive boosting approach for cross-project defect prediction." Softw Qual J Vol. 25, 2017, pp. 235–272.

[PS14] X.Y. Jing, F. Wu, X. Dong, and B Xu. "An Improved SDA Based Defect Prediction Framework for Both Within-Project and Cross-Project Class-Imbalance Problems." IEEE Transactions on Software Engineering, Vol. 43, No.4, 2017, pp. 321–339.

[PS15] C. Ni, W.S. Liu, X. Chen, Q. Gu, D.X. Chen, and Q.G. Huang. "A cluster-based feature selection method for cross-project software defect prediction." Journal of Computer Science and Technology, Vol. 32, No 6, 2017, pp. 1090-1107.

[PS16] X. Xia, D. Lo, S.J. Pan, N. Nagappan, and X. Wang. "HYDRA: Massively compositional model for cross-project defect prediction." IEEE Trans Softw Eng. Vol 42, 2016, pp. 977–998.

[PS17] S. Hosseini, B. Turhan, M. Mantyl. "Search based training data selection for cross project defect prediction." In: *ACM international conference proceeding series*. Association for Computing Machinery, New York, New York, USA, 2016, pp. 1–10.

[PS18] K. Kawata, S. Amasaki, and T. Yokogawa. "Improving relevancy filter methods for cross-project defect prediction." In *2015 3rd International Conference on Applied Computing and Information Technology/2nd International Conference*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

on Computational Science and Intelligence, IEEE. July, 2015, pp. 2-7.

[PS19] S. Herbold. "Training data selection for cross-project defect prediction." In: *ACM international conference proceeding series. Association for Computing Machinery*, 2013, pp 1–10

[PS20] J. Nam, J.S. Pan, S. Kim. "Transfer defect learning." In: *35th International conference on software engineering (ICSE).* 2013, pp 382–391.

[PS21} X. Jing, F. Wu, X. Dong, F. Qi, and B. Xu. "Heterogeneous cross-company defect prediction by unified metric representation and CCA-based transfer learning. In Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering, August 2015, pp. 496-507.

[PS22] Y. Zhang, D. Lo, X. Xia, and J. Sun. "An empirical study of classifier combination for cross-project defect prediction." In *2015 IEEE 39th Annual Computer Software and Applications Conference,* Vol. 2, July 2015, pp. 264-269.

[PS23] Z. Cai, L. Lu, and S. Qiu. "An abstract syntax tree encoding method for cross-project defect prediction." IEEE Access, Vol. 7, 2019, pp. 170844-170853.

[PS24] L. Sheng, L. Lu, and J. Lin. "An adversarial discriminative convolutional neural network for cross-project defect prediction." IEEE Access, Vol. 8, 2020, pp. 55241-55253.

[PS25] F. Wu, X.Y. Jing, Y. Sun, J. Sun, L. Huang, and F. Cui, and Y. Sun. "Cross-project and within-project semi supervised software defect prediction: A unified approach." IEEE Transactions on Reliability, Vol. 67, No.2, 2018, pp. 581-597.

[PS26] J. Nam, W. Fu, S. Kim, T. Menzies, and L. Tan. "Heterogeneous defect prediction. IEEE Transactions on Software Engineering," Vol. 44, No. 9, 2017, pp. 874-896.

[PS27] J. Chen, K. Hu, Y. Yang, Y. Liu, and Q. Xuan. "Collective transfer learning for defect prediction." Neurocomputing. Vol. 416, 2020, pp. 103-116

[PS28] R. Vashisht, and S.A.M. Rizvi. "Feature Extraction to Heterogeneous Cross Project Defect Prediction." In *2020 8th International Conference on Reliability*, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), IEEE, June, 2020, pp. 1221-1225.

[PS29] P.A. Habibi, V. Amrizal, and R.B. Bahaweres. "Cross-project defect prediction for web application using naive bayes (case study: Petstore web application)". In *2018 International Workshop on Big Data and Information Security (IWBIS)*, IEEE, May 2018, pp. 13-18.

[PS30] Xu, Z., Yuan, P., Zhang, T., Tang, Y., Li, S., & Xia, Z. HDA: Cross-project defect prediction via heterogeneous domain adaptation with dictionary learning." IEEE access, Vol. 6, 2018, pp. 57597-57613.

[PS31] L. Goel, M. Sharma, S.K. Khatri, and D. Damodaran. "Implementation of data sampling in class imbalance learning for cross project defect prediction." an empirical study. In 2018 *Fifth International Symposium on Innovation in Information and Communication Technology (ISIICT),* IEEE, 2018, pp. 1-6.

[PS32] S. Amasaki, K. Kawata, and T. Yokogawa. "Improving cross-project defect prediction methods with data simplification." In *2015 41st euromicro conference on software engineering and advanced applications*, IEEE, August, 2015, pp. 96-103.

[PS33] A. Bispo, R. Prudêncio, and D. Véras. "Instance Selection and Class Balancing Techniques for Cross Project Defect Prediction." In *2018 7th Brazilian Conference on Intelligent Systems (BRACIS),* IEEE, Oct. 2018, pp. 552-557.

[PS34] Y. Sun, X.Y. Jing, F. Wu, and Y. Sun. "Manifold embedded distribution adaptation for cross-project defect prediction." IET Software, Vol. 14, No. 7, 2021, pp. 825-838.