International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-4, November 2019

Performance Evaluation of Pseudo Code with Weka for Accuracy Calculation

Swaroop Potti, Dr. Prasanth Yalla, Venkata Naresh Mandhala, Saivardhan Pathuri, Subbaraju. B

Abstract*: Programming testing is a fundamental and essential advance of the existence cycle of programming improvement to recognize and defects in programming and afterward fix the deficiencies. The reliability of the data transmission or the quality of proper processing ,maintenance and retrieval of information to a server can be tested for some systems. Accuracy is also one factor that is usually used to the Joint Interoperability Test Command as a criterion for accessing interoperability. This is the main investigation of PC flaw forecast and exactness as per our examination, which spotlights on the utilization of PROMISE database dataset. Some PROMISE database dataset tests are compared between pseudo code (PYTHON) and actual software (WEKA),which in computer fault prediction and accuracy measurement are effective software metrics and machine learning methods.

Keywords*: Software Testing, Accuracy, Software defect Prediction, Machine Leaning Algorithm.

I. INTRODUCTION

Programming testing is a basic advance of the existence cycle of programming improvement which checks and approves the exhibition of programming. A lot of exertion is spent in the testing and investigating procedure to improve programming quality. However, designers for the most part have constrained assets and tight calendars as a rule, so they couldn't pay a lot for such a procedure. Defect detection methods were recommended to help organize code testing and investigating; they may propose segments of innovation which designers are probably going to be faulty. Fault expectation framework dependent on customary procedures is supplanted by AI based frameworks that can build framework upkeep execution. Efforts were made to think about disengagement of shortcomings, shirking of faults, identification of flaws and recuperation of faults. In this part, the analysts ' proposed approaches to PC deficiency expectation utilizing AI strategies were quickly examined. Machine learning is an artificial intelligence (AI) advancement that enables systems to pick up and develop subsequently in actuality without express programming. Machine learning focuses on PC programs that can get to data and use it to adapt separately. The learning method begins with observations and information, for instance, events, direct understanding, or analysis, to search for data examples and choose better decisions later on subject to the models they have. The basic goal is to normally empower PCs to learn without human intervention or help and to change exercises moreover. Calculations in AI are frequently characterized as directed and unaided.

In this paper, we laid a way for comparing software deficiencies and accuracy between pseudo code (PYTHON) and actual software (WEKA). This paper also provides an overview of measuring software deficiencies and accuracy for the PROMISE dataset. We considered 4 datasets, 3 of which were PROMISED data sets of NASA PORTAL and CAR company results. Open dataset, for example, "Apache POI" open source,[8]Rhino (an open-source JavaScript execution written in Java)[5] is utilized as trial information to test the model. Subsequently, a large portion of the informational collection utilized here is NASA dataset from the PROMISE database, which will focus on this examination. Guarantee database is intended for programming designing prescient models and information investigation.

Open dataset, for example, Open Source "Apache POI",[8]Rhino (an open-source execution of JavaScript written in Java) [5] are utilized as examination information to look at the model. In this manner, most dataset utilized in this field are NASA dataset from PROMISE archive that will be an attention on this audit. Guarantee storehouse is structure for prescient models and information examination in programming designing.

Software Reliability is the probability for estimating the errors in the stipulated term in any of the software systems. In our modern-day society, computers are used in various regions for different applications. For predicting and estimating software reliability there are many software reliabilities model.[19]

Manuscript published on November 30, 2019.
* Correspondence Author
Swaroop Potti*, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India.
Dr. Prasanth Yalla, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India. Email: prasanthyalla@kluniversity.in
Venkata Naresh Mandhala, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India. Email: mvnaresh.mca@gmail.com
Saivardhan Pathuri, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India.
Subbaraju. B, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India.

Promise Dataset: Guarantee database looks to give obvious and repeatable programming building examination models that can be utilized. Sharing informational collection among scientists and experts in programming building is an online store. An analyst or specialist can store their informational collection or other individuals can get to another model offer. Clients will subsequently have the option to think about their outcomes and close based on past works[7].

II. PREVIOUS RESEARCH

Impressive work was finished utilizing the product measurements AI calculation for the issue forecast process. A large portion of the examination that the NASA MDP (Metric Data Program) server utilized as the reference dataset. A dataset records all gadget issues found and the quantity of modules containing the shortcoming, though every module is characterized by a gathering of code level qualities or configuration level.

The most normally utilized dataset in flaw forecast studies is PROMISE database NASA databases that give a few parameters to be analyzed. It is prescribed to utilize the open dataset to render reproducible, refutable and undeniable issue expectation models. What's more, utilizing a similar open dataset will rearrange the investigation of correlation between one research and another. From a few Research that has been seen in this paper, programming metric is demonstrated to be one productive wellspring of prescient deficiency model.

Machine learning calculation is applied by ordering module into flawed and non-damaged models to consequently build issue prescient model. Execution based examination (exactness, resistance, accuracy, and working qualities of the collector) appeared from past trials, SVM and Random. Backwoods calculation gives two separate datasets the best expectation model. Such two calculations can be utilized for estimation of issue as a suggestion AI calculation. The vast majority of these examinations have exhibited utilizing most AI calculations to utilize programming metric for flaw expectation.

III. LITERATURE SURVEY

S.No	Author	Title	Journal	Research Findings
1	Benfano, Spits Warnars	Fault Prediction Using Machine Learning Approaches	2018	Breif Description of ML approaches
2	David Lo, Xin Xio	Deep Learning for Defect Prediction	2015	Deep learning implementation
3	Vandana Bhattarji	Software Fault Prediction Using Quad Tree-Based Clustering Algorithm	2016	K– Means,Quad Tree Implementation
4	Guru Prasad Bhandari, Ratneswar Gupta	Fault Predictability of Software using Deep Learning Techniques	December 2018	Deep learning implementation
5	Amod Kumar,Ashwni Bansal	Software Fault -Prediction Using GeneticBased Machine Learning Techniques	2017	Quad Tree Implementation
6	Tracy,David Bowes	The State of Machine-Learning inSoftware Fault Prediction	2016	ML Methodology
7	Joao R. Campos, Marco , Ernesto Costa	Exploratory Study of Machine Learning Techniquesfor Failure Prediction	2017	ML Techniques for failure Prediction
8	Laszo and Mukhejee	A Genetic Algorithm using Quad Trees	2016	ML Technique for failure Prediction
9	Wilsonn	Classifier Fitness on Accuracy. Evolutionary Computation	2015	Accuracy Calculation

10	Malhotra .T	Analysis of statistical and machine learning methods for predicting faulty modules	2014	Software Prediction faulty modules

IV. RELATED WORK

Fault detection system based on traditional techniques is replaced by AI-based systems that can increase system maintenance performance. Efforts were made to consider isolation of faults, avoidance of faults, identification of faults and recovery of faults. In this chapter, the researchers ' proposed approaches to computer fault prediction using machine learning methods were briefly discussed[4].

Machine Learning is firmly identified with machine measurements, which focus on PC based forecasts. Scientific advancement study furnishes the field of AI with procedures, hypothesis and application areas. Data mining is a field of concentrate inside AI and spotlights by solo learning on exploratory data assessment. Simulated intelligence is in like manner implied as judicious assessment in its application across over business issues.

ML is an approach which is centred on adapting naturally and enables PCs to develop and foresee the framework nature in view of past and the present disappointment information. Hence, it is very normal for programming specialists and analysts understand what specific technique is more significant to function admirably for a surrendered disappointment dataset and to what degree quantitatively [21].

A. Software Metrics

Software metric is a proportion of quantifiable or countable characteristics that can be used to measure and predict the quality of software. A metric is an indicator describing a specific feature of a software. Identifying and measuring software metrics is vital for various reasons, including estimating programming execution, measuring the effectiveness of software processes, estimating required efforts for processes, deduction of defects during software development, and monitoring and controlling software project executions. [18]

B. Confusion Matrix

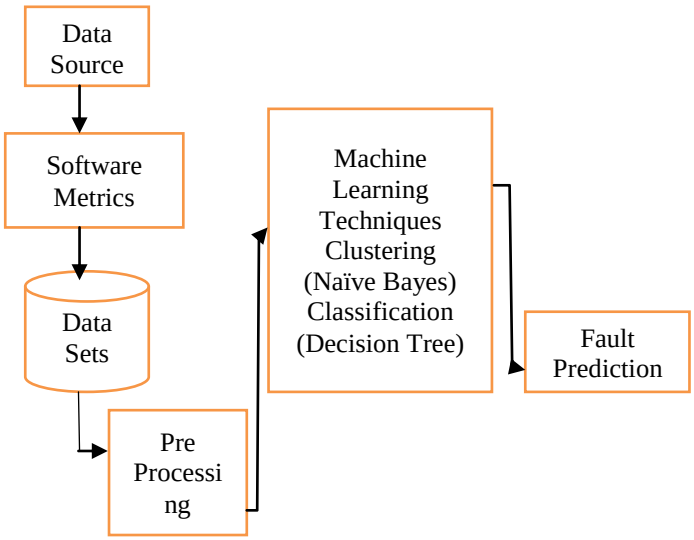
A confusion matrix is a table that is often used to define a characterization model's yield on an accumulation of test information characterized for the true values. It allows the visualization of an algorithm's output. It allows easy detection of ambiguity between classes, for example, one class is frequently mislabelled as the other. Many output measures are determined from the matrix of uncertainty. It also allows us to calculate precision, tolerance, memory and reliability.

C. Accuracy Calculation

	Classified	
	Yes (Predict)	No(Predict)
Yes(Resulted)	True Posi(TP)	False Neg(FN)
No (Resulted)	False Posi(FP)	Ture Neg(TN)

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN}$$

D. Process Flow



E. Clustering

It is basically a type of unregulated strategy for preparing. An unaided strategy for learning is a procedure by which correlations are taken from datasets containing data without named responses. It is routinely used as a gadget to find significant structure, central useful techniques, generative features, and groupings inborn in a social affair of models.

Grouping is the path toward separating the masses or data centres into different social affairs to make data centers in comparative get-togethers progressively like other data centers in a comparative get-together and specific from data concentrations in various get-togethers. It is basically an aggregation of things reliant on resemblances and separation between them. Clustering is significant on the grounds that it characterizes the inalienable gathering between the present unlabelled information. For an effective bunching, there are no conditions. This relies upon the buyer, which necessities they can use to address their issues. We might be fascinated, for example, in finding representatives of homogeneous social occasions (data decline), finding "customary bundles" and describing their dark properties ("run of the mill" data types), discovering accommodating and right groupings ("significant" data classes) or finding weird data objects (outer acknowledgment).The figuring will make a couple of assumptions that make centers unclear, and each supposition makes bundles that are exceptional and comparably genuine.

Clustering is the process of grouping the data into classes or clusters, so that objects within a cluster have high similarity in comparison to one another but are very dissimilar to objects in other clusters. Dissimilarities between objects can be calculated by the variety of attributes associated with the objects. In general clustering algorithm is categorized into Partition based, Hierarchical method, Density based methods, Grid based methods, Model based methods etc. In partition-based clustering k means clustering and its variations are most widely used [16].

Clustering is a procedure of finding similar data items (patterns, documents etc.) and then group the similar data together. Items belongs to different clusters are dissimilar data items, generally cluster values are considered as 1 or 0. The clustering process is not appropriate for all the cases sometimes these values are less than one. In practical situations clusters are not crisp, then it is represented as fuzzy. In order to enhance the clustering rate, two appropriate clustering approaches: K-means clustering and Fuzzy C Means (FCM) are considered. [20].

K-means clustering is considered to be a very famous and widely used partitioning based approach. It is a numerical, non-deterministic, an unsupervised and an iterative approach. In K-means clustering, the average value (mean value) of objects in the group represents individual cluster. The main objective of the K-means clustering algorithm is to acquire the stable number of clusters that reduces the Euclidean distances between data objects and centre of clusters [22].

F. Applications of Clustering in different fields

1. Promoting: It can be used to portray and discover customer sections for publicizing purposes.
2. Science It can be used for gathering among different kinds of plants and animals.
3. Libraries: It is used in gathering different books dependent on focuses and information.
4. Protection: It is used to perceive the customers, their game plans and recognizing the fakes.

G. Naive Bayes

This article discusses the speculation behind the Naive Bayes classifiers and their use. Straightforward Bayes classifiers are a gathering of portrayal computations reliant on Bayes' Theorem. It's definitely not a single figuring anyway a gathering of computations where all of them share an ordinary guideline, for instance each pair of features being described is self-sufficient of each other. Bayes Theorem finds the likelihood of an occasion happening given the likelihood of another occasion that has basically happened.CIT rushes to predict class of test instructive gathering. It further more perform-well in multi class conjecture. Exactly when assumption of opportunity holds, a Naive Bayes classifier performs better diverge from various models like key backslide and you need less planning data. It perform-well if there ought to emerge an event of full-scale data components diverged from numerical variable. For numerical variable, standard scattering is normal (ringer twist, which is a strong assumption).If hard and fast factor has an order (in test enlightening gathering), which was not seen in getting ready instructive accumulation, by then model will dole out a 0 (zero) probability and will be not capable make a figure. This is as often as possible known as "Zero Frequency". To get this, we can use the smoothing strategy. Here we are going to apply this technique by using python code and calculating accuracy for the PROMISED data set using pseudo code.

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other [17].

V. RESULTS

A. Classification

In AI and estimations, gathering is the issue of perceiving to which of a lot of portrayals (sub-masses) another acknowledgment has a spot, considering an availability set of information containing observations (or models) whose class backing is known. Models are allotting an offered email to the "spam" or "non-spam" class, and apportioning an assurance to a given patient dependent on watched attributes of the patient (sex, circulatory strain, closeness or nonappearance of unequivocal signs, and so forth.). Game-plan is an example of model certification.

```
classifier.fit(x_train,y_train)
y_pred=classifier.predict(x_test)
print(y_pred)
from sklearn.metrics import confusion_matrix
cm=confusion_matrix(y_test,y_pred)
print(cm)
from sklearn.metrics import accuracy_score
print("Accuracy Is ", accuracy_score(y_test,y_pred)*1
```

In the expressing of AI, gathering is viewed as an occasion of controlled learning, i.e., recognizing where a course of action set of reasonably apparent perceptions is accessible. The relating solo technique is known as bunching, and joins gathering information into classes dependent on some degree of regular likeness or division.

```
[[4.470e+02 8.260e+02 1.200e+01 ... 6.090e+02 1.550e+02 3.442e+03]
 [3.700e+01 2.900e+01 8.000e+00 ... 1.210e+02 3.800e+01 2.220e+02]
 [1.100e+01 4.050e+02 0.000e+00 ... 8.110e+02 4.110e+02 8.440e+02]
 ...
 [1.000e+01 7.000e+00 0.000e+00 ... 1.500e+01 1.800e+01 4.200e+01]
 [2.000e+00 1.000e+00 0.000e+00 ... 8.000e+00 9.000e+00 1.000e+01]
 [2.000e+00 5.000e+00 1.000e+00 ... 1.400e+01 1.200e+01 1.900e+01]]
['Y' 'Y' 'Y' ... 'N' 'N' 'N']
[0 0 0 ... 0 1 1]
[[1505   74]
 [ 272   68]]
Accuracy Is  81.96977592496091
```

B. Decision Tree

Choice tree computations are most commonly used counts in portrayal. Decision tree gives a successfully sensible showing technique and it furthermore unravels the request method.

Retrieval Number: D5394118419/2019©BEIESP
DOI:10.35940/ijrte.D5394.118419
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

The choice tree is clear instrument it encourages clients to look for after a tree structure reasonably so as to perceive how the choice is made in this segment essential point of view of choice tree strategies has been dissected with their qualities, preventions and applications. Within goal of choice tree is to make a model that figures the estimation of a basic variable dependent on various information factors. Regularly all choice tree figuring are made in two stages (I) tree headway; in which preparing set dependent on near to consummate criteria is isolating recursively until a gigantic piece of the record having a spot with the segment having same class name (ii) tree pruning; in which size of tree is lessened making it continuously clear . Here we will concentrate on ID3 and choice tree figuring. ID3 (IterativeDichotomiser3) choice tree figuring was displayed in 1986. It is one of the broadly utilized calculations in the district of information mining and AI because of its adequacy and straightforwardness. The ID3 tally depends upon data gain. A piece of the qualities and shortcomings of ID3 choice tree are appeared in attributes wires; direct and in genuine end entire preparing model is considered while inadequacies combine; no back after pursuing, unfit to oversee missing attributes and no general improvement.C4.5 is an extraordinary figuring for choice trees creation. It is an improvement of the ID3 estimation and Classification Techniques in Machine Learning Journal of Basic and Applied Sciences it purposes of containments its disadvantages accomplished by ID3. In pruning stage C4.5 attempts to butcher the un-comfort branches by swapping them with leaf focus indicates by restoring the tree once it has been made. The attributes of C4.5 are administering preparing information with missing portion respects, bargains both discrete and enduring highlights and giving office of both pre and post pruning.[23].

```
classifier=DecisionTreeClassifier(criterion='entropy',
random_state=0)
classifier.fit(x_train,y_train)
y_pred=classifier.predict(x_test)
print(y_pred)
from sklearn.metrics import confusion_matrix
cm=confusion_matrix(y_test,y_pred)
print(cm)
from sklearn.metrics import accuracy_score
print("Accuracy Is ",
 accuracy_score(y_test,y_pred)*100.0)
```

```
[[4.470e+02 8.260e+02 1.200e+01 ... 6.090e+02 1.550e+02 3.442e+03]
 [3.700e+01 2.900e+01 8.000e+00 ... 1.210e+02 3.800e+01 2.220e+02]
 [1.100e+01 4.050e+02 0.000e+00 ... 8.110e+02 4.110e+02 8.440e+02]
 ...
 [1.000e+01 7.000e+00 0.000e+00 ... 1.500e+01 1.800e+01 4.200e+01]
 [2.000e+00 1.000e+00 0.000e+00 ... 8.000e+00 9.000e+00 1.000e+01]
 [2.000e+00 5.000e+00 1.000e+00 ... 1.400e+01 1.200e+01 1.900e+01]]
['Y' 'Y' 'Y' ... 'N' 'N' 'N']
[0 0 0 ... 0 0 0]
[[1365  214]
 [ 210  130]]
Accuracy Is  77.90515893694632
```

C. Comparative Analysis

	Naïve Bayes	Decision Tree
Nasa Dataset (Module 1)	81	77
Nasa Dataset (Module 2)	80	76
Car Company (DS3)	69	80
Student List (DS4)	77	86

So, From the above table we can conclude that clustering gives better accuracy for bigger data sets and classification gives better results for smaller datasets. Machine Learning calculation is applied to consequently develop deficiency prescient model by grouping module into damaged and non-imperfect one. These two calculations can be utilized as a suggestion AI calculation for flaw forecast for better accuracy and precision. By using these techniques, we can estimate the fault prediction and accuracy of the given data. In some cases, classification gives more accuracy than naïve Bayes when comes to smaller datasets

D. Weka Software

Weka contains a social event of acknowledgment contraptions and calculations for information appraisal and farsighted representing, together with graphical UIs for direct access to these limits. The first non-Java form of Weka was a TC l/Tk front-end to (generally distant) demonstrating estimations acknowledged in other programming vernaculars, despite information preprocessing utilities in C, and a Make report-based framework for running AI tests. This unique rendition was in a general sense composed as a contraption for isolating information from agrarian territories, in any case, the later absolutely Java-based structure (Weka 3), for which improvement began in 1997, is before long utilized in various application areas, unequivocally for useful purposes and research.

Advantages

Free availability under the GNU General Public License.

Convenient, since it is totally executed in the Java programming language and thusly continues running on basically any front-line figuring stage.A thorough accumulation of information preprocessing and demonstrating procedures.

Usability because of its graphical UIs.

E. Native regression tools

Weka has countless relapse and order instruments. Local bundles are the ones incorporated into the executable Weka programming, while other non-local ones can be downloaded and utilized inside R. Weka condition. Among the nearby packages, the most outstanding device is the M5p model tree group. The full rundown of mechanical assemblies is open here.

F. Decision Tree

```
Root mean squared error                  0.4857
ative absolute error                    80.8089 %
t relative squared error                125.5084 %
Total Number of Instances             9593

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Ar
               0.357     0.149     0.350      0.357     0.354      0.207    0.591     0.248
               0.851     0.643     0.855      0.851     0.853      0.207    0.591     0.844
Weighted Avg.  0.761     0.552     0.762      0.761     0.761      0.207    0.591     0.735

=== Confusion Matrix ===

    a    b   <-- classified as
  628 1131 |    a = Y
 1166 6668 |    b = N
```

G. Bayes Theorem

```
ean absolute error                      0.1863
oot mean squared error                  0.4291
elative absolute error                 62.2139 %
oot relative squared error            110.8776 %
otal Number of Instances              9593

== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Cl
               0.205     0.049     0.484      0.205     0.288      0.226    0.678     0.340     Y
               0.951     0.795     0.842      0.951     0.893      0.226    0.678     0.892     N
eighted Avg.   0.814     0.658     0.776      0.814     0.782      0.226    0.678     0.791

== Confusion Matrix ===

    a    b   <-- classified as
  361 1398 |    a = Y
  385 7449 |    b = N
```

H. Comparative Analysis

	Naive Bayes (CODE)	Decision Tree (CODE)	Naïve Bayes (WEKA)	Decision Tree (WEKA)
Nasa Dataset (Module 1)	81	77	83	79
Nasa Dataset (Module 2)	80	76	83	80
Car Company (DS3)	69	80	73	84
Student List (DS4)	77	86	80	89

From the above table, we can clearly tell that accuracy predicted using Weka tool gives better accuracy. But even the pseudo code provided also gives somewhat near values to the software.

VI. CONCLUSION

Through this project we implement Machine learning operations using the pseudo code for fault prediction of data sets which we usually do with the Weka tool. The two pseudo codes one for clustering and other for classification are implemented using decision tree and naïve Bayes algorithms. Our code produces the closest predictions to accurate. However, Weka tool predictions will be a little contrasting to the pseudo code predictions.

REFERENCES

1. G Benfano Soewito Computer Science Department, Software Metrics for Fault Prediction Using Machine Learning Approaches – Master of Computer Science,Bina Nusantara University Jakarta, Indonesia,2017
2. Xinli Yang∗, David Lo,Deep Learning for Just-In-Time Defect Prediction,College of Computer Science and Technology, Zhejiang University, Hangzhou, China School of Information Systems, Singapore Management University, Singapore,2017
3. VandanaBhatarji,Software Fault Prediction Using Quad Tree-Based K-Means Clustering Algorithm, Consiece paper,2018
4. Ratneshwer Gupta,Measuring the Fault Predictability of Software using Deep Learning Techniques with Software Metrics,Jawaharlal Nehru University New Delhi, India,2018
5. G. Boetticher, T. Menzies, T. Ostrand, PROMISE Repository of Empirical Software Engineering Data, West Virginia University, Department of Computer Science, 2007.
6. Fenton, N. E., & Pfleeger, S. L. (2009). Software metrics: a rigorous and practical approach. Methods, 5, 26.
7. G. Boetticher, T. Menzies, T. Ostrand, PROMISE Repository of Empirical Software Engineering Data, West Virginia University, Department of Computer Science, 2007.
8. R. Malhotra and A. Jain, "Fault prediction using statistical and machine learning methods for improving software quality," J. Inf. Process. Syst., vol. 8, no. 2, pp. 241–262, 2012.
9. M. Laszlo and S. Mukherjee, "A Genetic Algorithm Using Hyper-Quadtrees for Low-Dimensional K-Means Clustering," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 28, no. 4, pp. 533-543, Apr. 2006.
10. Kovacs T., Genetics-based machine learning. In Handbook of Natural Computation: Theory, Experiments, and Applications. Springer-Verlag, 2011.
11. Bull L., Kovacs T., Foundations of Learning Classifier Systems: An Introduction, 2005, Stud-Fuzz 183, 1–17 (2005),Springer.
12. Wilson S. W., ZCS: A zeroth level classifier system. Evolutionary Computation, 2(1):1–18, 1994 [13]. Wilson S. W., Classifier Fitness Based on Accuracy. Evolutionary Computation, 3(2):149–175, 1995.
13. Wilson S. W. Generalization in the XCS classifier system. Genetic Programming 1998: Proceedings of the Third Annual Conference, pages 665–674. Morgan Kaufmann, 1998.
14. Jiang Y, CukicB, Menzies T. Cost curve evaluation of fault prediction models. In: 19th International Symposium on Software Reliability Engineering. Seattle; 2008. p. 197-206.
15. Nirmal, K. R., & Satyanarayana, K. V. V. (2016). Issues of K means clustering while migrating to map reduce paradigm with big data: A survey. International Journal of Electrical and Computer Engineering, 6(6), 3047-3051. doi:10.11591/ijece.V6i6.11207
16. Banchhor, C., &Srinivasu, N. (2016). CNB-MRF: Adapting correlative naive bayes classifier and MapReduceframework for big data classification. International Review on Computers and Software, 11(11), 1007-1015.doi:10.15866/irecos. v11i11.10116
17. Prasanth, Y., Sreedevi, E., Gayathri, N., & Rahul, A. S. (2017). Analysis and implementation of ensemble feature selection to improve accuracy of software defect detection model. Journal of Advanced Research in Dynamical and Control Systems, 9(18 Special Issue), 601-613. Retrieved from www.scopus.com
18. Prasanth, Y., Grace Hepsiba, M., Mounika, T., Pavan Kumar, T., & Raghavendra Kumar, G. (2017). Application of machine learning techniques on naval and telecommunication system failure data. Journal of Advanced Research in Dynamical and Control Systems, 9(Special Issue 6), 1065-1076. Retrieved from www.scopus.com.
19. Ramesh, C. R., Rao, K. R., & Jena, G. (2018). Fuzzy clustering algorithm efficient implementation using centre of centres. International Journal of Intelligent Engineering and Systems, 11(5), 1-10. doi:10.22266/IJIES2018.1031.01.

20. Krishna Mohan, G., Yoshitha, N., Lavanya, M. L. N., & Krishna Priya, A. (2018). Assessment and analysis of software reliability using machine learning techniques. International Journal of Engineering and Technology(UAE), 7(2.32 Special Issue 32), 201-205. Retrieved from www.scopus.com.
21. Rachapudi, V., Venkata Suryanarayana, S., &SubhaMastan Rao, T. (2019). Auto-encoder based K-means clustering algorithm. International Journal of Innovative Technology and Exploring Engineering, 8(5), 1223-1226. Retrieved from www.scopus.com.
22. Mohiddin, S. K., Kumar, P. S., Sai, S. A. M., &Santhi, M. V. B. T. (2019). Machine learning techniques to improve the results of student performance. International Journal of Innovative Technology and Exploring Engineering, 8(5), 590-594. Retrieved from www.scopus.com.

AUTHORS PROFILE

Prasanth Yalla received his B.Tech Degree from Acharya Nagarjuna University, Guntur (Dist), India in 2001, M.Tech degree in Computer Science and Engineering from Acharya Nagarjuna University in 2004, and received his Ph.D. degree in CSE titled "A Generic Framework to identify and execute functional test cases for services based on Web Service Description Language" from Acharya Nagarjuna University, Guntur (Dist), India in April 2013. He was an associate professor, with Department of Information Science and Technology in KL University, from 2004 to 2010. Later he worked as Associate professor, with the department of Freshman Engineering from 2011 in KL University. Presently he is working as Professor in the department of Computer Science & Engineering in KL University and also Associate Dean (R&D) looking after the faculty publications. Till now he has published 28 papers in various international journals and 4 papers in conferences. His research interests include Software Engineering, Web services and SOA. He taught several subjects like Multimedia technologies, Distributed Systems, Advanced Software Engineering, Object Oriented Analysis and design, C programming, Object-Oriented programming with C++, Operating Systems , Database management systems, UML etc. He is the Life member of CSI and received "Active Participation- Young Member" Award on 13-12-13 from CSI. He has applied a project to SERB very recently.