IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023

DOI: 10.17148/IJARCCE.2023.12679

Implement Classification Approach for Software Defect Prediction

Asst. Prof. Suraj Yadav[1], Keertika Sirohiya[2]

Assistant Professor, Jagannath University, Jaipur, Rajasthan, India[1]

M.Tech. Researcher, Department of Computer Science, Jagannath University, Jaipur[2]

Abstract: For any specific kind of application software is designed by considering its platforms related to hardware and operating system. Before initiating the development of software products, it is important to plan the process carefully. A detailed knowledge related to the actual requirement of the user and he processes to be performed when developing the software are must since the software is large in size. To generate the source code from a sequential model, the software engineering based abstract-present model is applied for generating the code of secondary phase, the code designed from one phase is given as input to design another code. For classifying the required and non-required code through which the next phase of code can be designed, the SVM classifier is applied. Python is used to implement the proposed model. Based on certain performance parameters, the results are evaluated in this research.

Keywords: Software Development, SVM Classifier, Reverse Engineering, Python Code, Software Design, Python Classifier, SVM Approach, Application Software.

I. INTRODUCTION

Software Engineering is known as a method that is followed systematically to design the software. It is important to have certain guidelines while designing the software since it is complex. Software is applied with the objective of achieving certain tasks. Including single program may or may not be possible. The different properties and procedures of various techniques help in designing a highly efficient software product. The software can be categorized among system software and application software. All the hardware components are managed and then used as functional module by the system software. For any specific constraint, the software product can be designed and few specified operations can be executed in case of application software. The software and program are very different from each other. Since source code and object code are combined, the program is subset of software. An executable code that serves certain computational purpose is known as a program and when programs are collectively applied, the software is designed. A component software product is designed based on the properties of {various procedures and techniques provided by software engineering. Therefore, before designing any software product, it is important for the software developer to consider the time, budge and cost.

II. LITRATURE REVIEW

Senay Tuna Demirel in their research paper (2018) studied that a major challenge in different projects of software development was identified as requirement analysis. Client prerequisite design and running caused several effects to software task. Hence, a lot of research works were needed to make enhancements in both studious and manufacturing domains. Some models such as CMMI revealed prerequisite improvement and organization. These models also identified the particular aims and practiced to achieve these aims. The listing of major challenges and concerns of requirement management was done in this work regarding standardization activity called as CMMI.

Syed Waqas Ali in (2018) studied that it was essential to ensure the quality of Software Requirement Specification (SRS) for improving the quality of software products. In this work, a novel approach had been recommended for resolving several issues affecting the SRS. Every stage of the proposed approach worked on its own limits. This approach could also decrease the negative aspects in efficient manner. These aspects affected the quality of soft projects in direct or indirect manner. . It was advantageous to inspect the external party followed by making certain the last stipulation of requirements mapping. In general, this event made judgment about several quality aspects of Software Requirement Specification (SRS). This procedure generally improved the document's quality. This factor directly affected the development of the project in positive way.

Marc Fyrbiak in 2018 provided an overview of HRE (Hardware Reverse Engineering) as well as its various existing challenges. This work was carried out in three stages. In the first stage, a systematic analysis of recent research subdivisions from decapsulation to gate-level netlist regarding HRE was carried out in this work. On the basis of analysis,

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023

DOI: 10.17148/IJARCCE.2023.12679

the formulation of various openly existing research queries was carried out for the quantification of reverse engineering (RE) in technical manner. In the next stage, a lot of efforts were made for resolving different issues and acquiring knowledge. Also, a discussion had been made on its ability to measure human factors in RE (Reverse Engineering). Thirdly, new solutions were suggested for upcoming interdisciplinary study that included both technological as well as emotional viewpoints. These viewpoints assured to completely found out the complications of HRE (Hardware Reverse Engineering).

Paula Fraga-Lamas **(2017)** identified the contribution of RFID (Radio Frequency Identification) in the evolution of several applications related to public transport. This work offered a new technique for RE (reverse engineering). The recommended approach also identified the different flaws in safety applications. In particular, this work analyzed the communication protocols of a public transport card. This card had been used by multiple Spanish users. A hardware device called Proxmark 3 was implemented with the recommended technique in this work. The access of personal data could be provided by this approach. The main aim here was to get hold of messaging between tag and reader. This approach also emulated both tags as well as readers.

Tuan Anh Nguyen in (2017) studied that it would be important for the developers of industrial software to pinpoint some groups implementing appealing real-time ideas in an object-based program. In this work, the encryption of two interpretations about the performance of programmer was carried out in Reoom. It was a new frivolous stationary study. This tool was compared with its very close competitor called Womble on third-party open source applications. The achieved results revealed that that Reoom was used for much wider purposes. This tool outperformed another tool in terms of recall value and general accuracy.

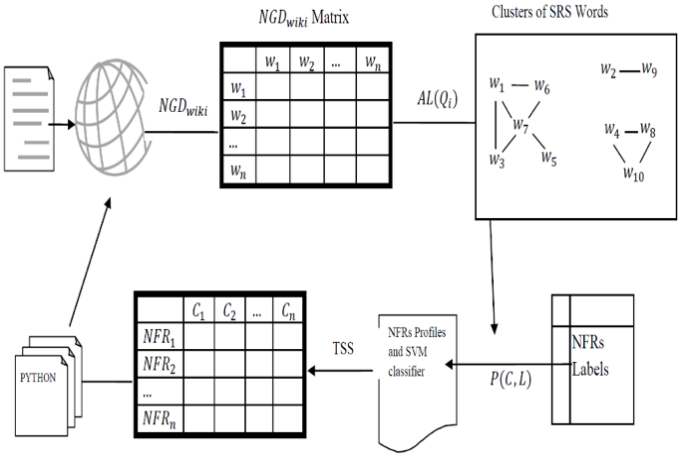
Research Objectives:

The current study has been conducted to accomplish the following research objectives as:

1. To implement LTSA-SVM approach for predicting defects in software.
2. To design software defect forecasting paradigm on the basis of linear SVM (Support Vector Machine)
3. To implement proposed system and compared with existing LTSA0SVM LLE-SVM algorithm in terms of different metrices.

III. RESEARCH METHODOLOGY

In the primary stage of analysis, the identification of possible non-functional requirements (NFRs) existing within a system will be carried out. In this stage, research theories will be generated on the basis of valuable hypothesis of cloud. The documents with similar behaviour occurring in the same cluster on the basis of the relatedness of needed data. This performance (behaviour) is very important to the knowledge about particular words of exiting content. Hence, particular words will be grouped in same cluster. On the basis of this technique, the implementation of certain suppositions will be carried out. The key objective of cluster analysis is to extract keywords from operational needs of software system. The semantically coherent groups of natural language words will be generated using these key words. In this approach, the clustering of a set of separate groups including same words will be carried out in semantic manner. The description of theoretical themes having the ability to pass the original test will be done in ideal manner with the help of clusters in C. The quality limitations of a system can be represented efficiently using these themes. This works represents NP-hard issue that identifies the optimum structures of a cluster. These structures are fitted in most favourable manner.



IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

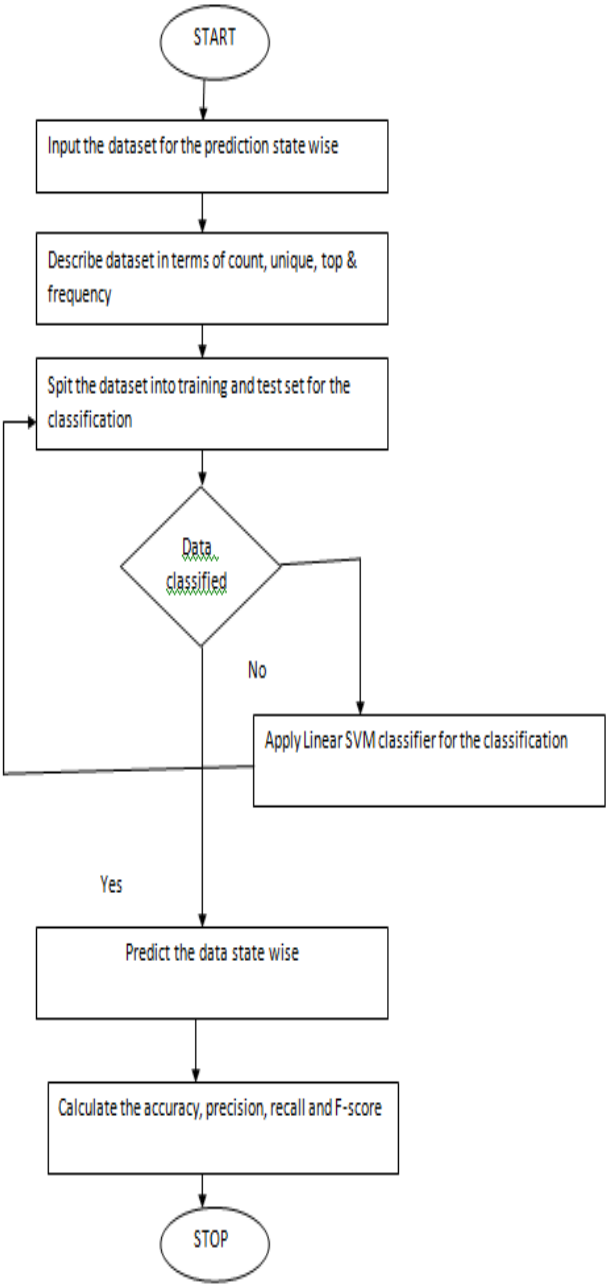
International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023

DOI: 10.17148/IJARCCE.2023.12679

On the other hand, some tests will be conducted for providing almost best solutions. The evaluation of sequence of semantic similar measures will be carried out with the word clustering techniques for determining these types of structures. As shown in figure 3.1, the recommended scheme is implemented with the help of several steps. The cluster quality objective function is represented by Qi in this figure. At the same time, $P(C,L)$ gives the formula of classification. AL denotes the Average Linkage clustering approach while TSS denotes test semantic similarity. This work implements linear SVM for carrying out classification. The classification is reduced using the verification (validation) set. This work classifies the acceptable and non-acceptable function using the CM1 dataset. The implementation of NFR matrix is carried out along with the linear SVM (Support Vector Machine) classification model. The comparison of recommended and available approaches is carried out in terms of certain metrics for evaluating the obtained enhancements. The implementation of NFR matrix will be done for classifying data.

The key purpose of recommended technique is to carry out classification. In order to fulfil this aim, this algorithm combines NFR matrix with linear SVM approach.



IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023

DOI: 10.17148/IJARCCE.2023.12679

Pseudo Code

Input

Training Dataset T

Output

A class of Testing Dataset

Steps

1. Read the Training Dataset T;
2. Extract features of the Training dataset T;
3. Apply cross validation for the data division;
4. Divide Training set into training and testing data;
5. Calculate the likelihood for each class
6. Get the greatest likelihood
7. Apply Linear SVM Classifier

    7.1 Read the training dataset T

    7.2 For each instance value in the test set do

        7.2.1 for i,j=1 to m do

            If $D(SV_j<SV_i)<$hyperplane

                $SV_j<-SV_i$

                Update SV_j

            Else

                Update Svi

            End if

        End for

    End for

IV. PROPOSED WORK AND PROBLEM FORMULATION

Reverse engineering can be defined as a technique that generates code from the outlined UML model. An important role has been played by sequence diagram for generating model. This is done to derive the competent code by means of this approach. A paradigm of absent-present is proposed to implemented in the work for reverse engineering. First of all, the analysis of a source code in its primary state will be carried out to generate the source code of next phase. In this paradigm, the syntax code will be generated by developing the abstract-syntax tree. The venture considers sequence diagrams to derive the source code of novel stage from sequence diagram. Identifying the necessary source code to carry out classification will tend to be the key purpose here. This phenomenon enhances the trustworthiness of the developed paradigm. This work will try to fulfil the gaps of earlier research works. Two vital works are:

1. The classification of sequence diagram is important.
2. The existing techniques are the combination of feature reduction and classification.

V. CONCLUSION

This work concludes that RE (Reverse Engineering) is an extremely effective approach of source code origination. The base paper makes use of abstract-present paradigm to generate the source code. The implementation of the earlier sequence's code is carried out in the abstract-present paradigm for this purpose. Here, different methods such as extraction, abstraction and visualization are also implemented. In this work, the implementation of classification algorithm is done before the visualization. This phenomenon enhances the trustworthiness of abstract-present paradigm based on RE. This work makes use of python for implementing the recommended approach. Certain performance criteria are used in this work for analysing the achieved outcomes. The recommended approach is based on linear SVM.

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023
DOI: 10.17148/IJARCCE.2023.12679

VI. SUGGESTIONS

1. The comparison of recommended approach with other state-of-the-art techniques of SDP (Software Defect Prediction) for verifying outcomes.
2. Use of clustering and data set simplification approach for making improvements in the recommended approach.

REFERENCES

[1] Jirayus Jiarpakdee, Chakkrit Tantithamthavorn, Akinori Ihara, Kenichi Matsumoto, "A Study of Redundant Metrics in Defect Prediction Datasets", IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), volume 5, issue 14, pp- 1937-1946, 2016.
[2] Chakkrit Tantithamthavorn, Shane McIntosh, Ahmed E. Hassan, Kenichi Matsumoto, "Comments on Researcher Bias: The Use of Machine Learning in Software Defect Prediction", IEEE Transactions on Software Engineering, Volume: 42, Issue: 11, pp- 758-764, 2016.
[3] B. Shakya, Mark M. Tehranipoor, Swarup Bhunia, Domenic Forte, "Introduction to Hardware Obfuscation: Motivation, Methods and Evaluation", Hardware Protection through Obfuscation, Springer, ch. 1, pp. 3–32, 2017.
[4] Shahed E. Quadir, Junlin Chen, Mark Mohammad Tehranipoor, "A survey on chip to system reverse engineering", Journal on Emerging Technologies in Computing Systems, vol. 13, no. 1, pp. 6:1–6:34, 2016.
[5] Pramod Subramanyan, Nestan Tsiskaridze, Wenchao Li, Adrià Gascón, Wei Yang Tan, Ashish Tiwari, "Reverse Engineering Digital Circuits Using Structural and Functional Analyses", IEEE Trans. Emerging Topics Computing, vol. 2, no. 1, pp. 63–80, 2014.
[6] U. Guin, Ke Huang, Daniel DiMase, John M. Carulli, Mohammad Tehranipoor, Yiorgos Makris, "Counterfeit Integrated Circuits: A Rising Threat in the Global Semiconductor Supply Chain," Proceedings of the IEEE, vol. 102, no. 8, pp. 1207–1228, 2014.

