International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8, Issue-1, May 2019

Optimization of the Bug Report Classification using Genetic Algorithm

Gayathri P M, Greeshma K Babu, Deepa G

Abstract: ***A bug report is an effective way of communicating the bugs among bug reporters and bug recipients. At the same time ,bad bug reports are long, inefficient form of communication for all concerned and do not contain relevant information to resolve the problems .The misclassification in bug report is therefore a serious issue that scarifies the accuracy of bug reports. Here we propose an approach to merging text mining and NLP to identify bugs and nonbugs in a bug report. In this system, KNN and Info Gain are used to classify and Genetic algorithm are used to optimize and improve automatic bug prediction performance.***

Index Terms: KNN (K Nearest Neighbour) classifier, NLTK (Natural Language Tool Kit), Genetic algorithm.

I. INTRODUCTION

Software bug prediction has boomed in the software development and maintenance phase in recent years. It predicts different aspects such as severity, priority, location or number based on repository data. Incorrect information is moreproblematicthanlackofinformation, bugreportsbeing the most effective way of knowing the bugs present in the project, it is necessary that it must be correct and relevant .So misclassification is performed using KNN and Info Gain. Weproposed that usage of genetic algorithm along with lassifierscanimprovetheaccuracyandtherebyoptimisethe result ofclassification.

II. METHODSUSEDINPROPOSEDSYSTEM

A. Tokenization

Tokenization is splitting of larger sentences into smaller words. There are various methods of tokenization such as n – gram tokenizer, alphabetic tokenizer and word tokenizer.

Example:

Input: she is so beautiful Output: she, is, so, beautiful

B. Stop -wordsRemoval

A stop word is a commonly used word such as the, a, an, in that a search engine has been programmed to avoid.

C. Stemming

Stemming is a technique used to trim the derived terms to theirrootword.Therearesomestemmingalgorithmssuchas Table-lookup approach, n-gram stemmer, Successor variety, Affix Removal stemmeretc.

D. TF-IDF

Term Frequency and Inverse Document Frequency are intentional numerical statistics to reflect theimportance of a word for a document in a collection.

E. Bigram

Abigramisaseriesoftwoadjacentelementsfromastringof tokens, which are typically letters orwords.

Example:

Before applying bigram It is, not, error.

After applying bigram It is ,not error.

F. GeneticAlgorithm

Genetic algorithm is natural evolution inspired heuristic search and optimization technique.GA makes slight changes to its solutions slowly until getting the best solution, and is used to create high quality solutions based on bio inspired operators such as mutation, crossover and selection.

G. InfoGain

Information gain counts the number of bits of information obtained by knowing the presence or absence of a term in a document.

H. KNNClassifier

K nearest neighbors (KNN) is an algorithm that stores all convenient cases and classifies new cases according to a measure of similarity. It works to determine the nearest k neighbors based on the minimum distance from the query instance to the training samples.

Revised Manuscript Received on 30 May 2019.

***** Correspondence Author

Gayathri P M, Computer Science and IT, AmritaVishwaVidyapeetham/ Amrita School of Arts and Scineces Kochi, Ernakulam, India.

Greeshma K Babu, Computer Science and IT, Amrita VishwaVidyapeetham/ Amrita School of Arts and Scineces Kochi, Ernakulam, India.

DeepaG,ComputerScienceandIT,AmritaVishwaVidyapeetham/Amrita School of Artsand Scineces Kochi, Ernakulam, India.

III. RELATEDWORK

AshimaKukkar, Rajini Mohan [1] proposed a hybrid approach to identify which bug report as bug or non bugs by merging text mining, natural language processing and machine learning techniques. After applying preprocessing technique like tokenization ,stop words removal, stemming
,bigram and TF-IDF are applied on bug reports .Bigram approach helped to reduce the dataset sparsity.The feature extraction and feature selection method are applied in this work Info- Gain algorithm and KNN is utilized.

IV. PROPOSEDSYSTEM

From extensive literature survey, it is concluded that the misclassification in bug reports is a need to be solved. In this paper we propose a technique to tackle the problem of misclassification by using KNN classifier and info-gain. Work flow of proposed system is depict in fig1 .In this paper we propose that incorporating genetic algorithm with KNN and Info Gain to optimizes the result and thereby improve the accuracy. The primary process is data collection and we have collected data related to eclipse. We have taken the summary, bugid, label and other attributes like severity, priority,statusandcomponentsofbugreportsfor processing. Later on, pre-processing steps like tokenization, stop words removal and stemming are applied on textual bug summary. After preprocessing feature extraction is done using Term frequency- inverse Document Frequency (TF-IDF) to calculate frequency scores. TF-IDF feature extraction technique helps to find the occurrence of feature in document,andthenbigrammethodisappliedtocombinethe two adjacent words. Then the genetic algorithm is used for optimizing the data after preprocessing the optimized results are the passed to Info-gain and KNN to classify the bug reports and distinguish them as bugs and non bugs, the proposed system is implemented using python. The main advantage of including genetic algorithm is to get an optimized solution and thereby improving accuracy of system.

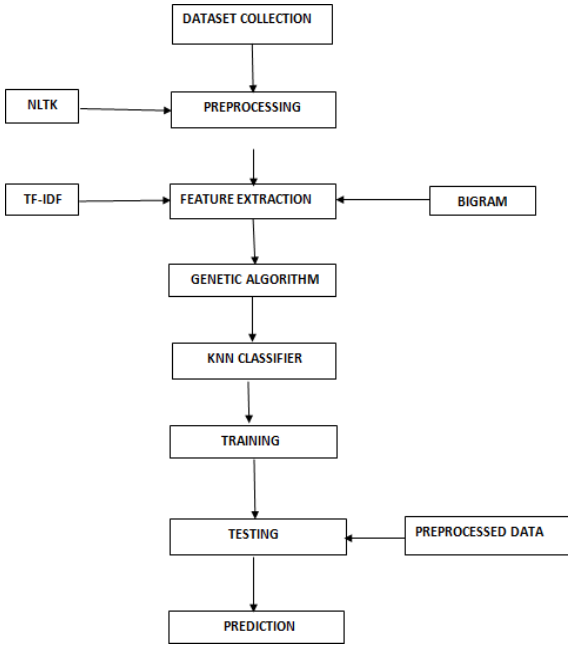

Fig : 1 Proposed System

V. EXPERIMENT ANDRESULT

In this paper bug report classification is being done on about huge bug data in eclipse and after the implementation it is being found that the accuracy of the system is improved to 92.02%. By applying genetic algorithm for optimization the accuracy of the system is being measured based on the inference obtained from confusion matrix.

VI. CONCLUSION AND FUTUREWORK

ABugreporthelpsthedevelopersin identifyingthebugsand resolving them. But often misclassification is an issue that scarifiestheaccuracyofbugreports,sointhispaperwehave proposed an approach to predict the bugs and no bugs within bug report, and also proposed genetic algorithm to optimize the result and thereby improve the accuracy and we got the accuracyrateas92.02%basedon confusion matrix.Wehave trained the supervised bug report, for future work we suggest to implement the same methodology of bug report classification and optimization on unsupervised bugreport.

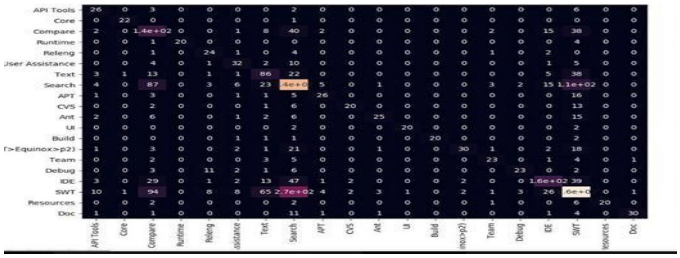

Fig:2 Confusion Matrix.

REFERENCES

1. AshimaKukkara, RajniMohana: A Supervised Bug Report Classification with Incorporate andTextualfieldKnowledge. International ConferenceonComputationalIntelligenceandDataScience(ICCIDS2018)
2. KanchanRawat, Dr. Ajay Goel: A Novel Hybrid Approach towards an Improvement in Bug Severity prediction for Better Software Quality. International Journal of Engineering Trends and Technology (IJETT) – Volume 38 Number 2- August(2016).
3. GitikaSharmaa,,SumitSharmaa, ShrutiGujrala: A Novel Way of Assessing Software Bug Severity Using Dictionary of Critical Terms. 4th International Conference on Eco-friendly Computing and Communication Systems, ICECCS(2015).
4. ArjunchandraC,SandhyaCJandDeepaG:AMethodForImprovingThe AccuracyOfBugMiningByReplacingStemmingWithLemmatization.
5. Divyavarma K, RemyaM, DeepaG: An Enhanced Bug Mining for Identifying Frequent Bug Pattern using Word Tokenizer and FP-Growth.AdvancesinIntelligentSystemsand Computing.515,pp.525-532.(2017).
6. Prabhsharan Kaur1, Charanjeet Singh2: A Systematic Approach for Bug Severity Classification using Machine Learning's Text Mining Techniques. International Journal of Computer Science and Mobile Computing, Vol.5 Issue.7, July- (2016), pg.523-528.
7. Nikitak.Thool,EktaB.Kumari,KiranL.Soni,VaishnaviG.Dhande,Prof. Sunny G. Gandhi: Bug classification system using data mining algorithms. InternationalJournalofResearchInScience&EngineeringVolume3Issue1 Jan-Feb(2017).
8. K.Herzig,S.Just,andA.Zeller(2013):"It'snotabug,it'safeature:How misclassification impacts bug prediction," in Proceedings of the 2013 InternationalConferenceonSoftwareEngineering.IEEEPress,pp.392–401.S.
9. M.D'Ambros,M.Lanza,andR.Robbes(2010):"Anextensivecomparison of bug prediction approaches," in Mining Software Repositories (MSR), Working Conference on. IEEE, pp.31–41.
10. ZhouY,TongY,GuR,GallH(2016):Combiningtextminginganddata miningforbugreportclassification.JSoftwEvolProcess28(3):150–176.
11. Lamkanfi A, DemeyerS, Giger E, Goethals B (2010) :Predicting the severity of a reported bug. In: Proceedings of the 2010 7th IEEE working conferenceonminingsoftwarerepositories(MSR'10).IEEE,pp1–10.