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 more problematic than lack of information, bug reports being 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. We proposed that usage of genetic algorithm along with classifiers can improve the accuracy and thereby optimise the result of classification.

II. METHODS USED IN PROPOSED SYSTEM

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 -words Removal

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 their root word. There are some stemming algorithms such as Table-lookup approach, n-gram stemmer, Successor variety, Affix Removal stemmer etc.

D. TF-IDF

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

E. Bigram

Bigram is a series of two adjacent elements from a string of tokens, which are typically letters or words.

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. KNN Classifier

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.
III. RELATED WORK
Ashima Kukkar, 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. PROPOSED SYSTEM
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, status and component of bug reports for processing. Later on, pre-processing steps like tokenization, stop words removal and stemming are applied on textural 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 and then bigram method is applied to combineth the 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.

V. EXPERIMENT AND RESULT
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 FUTURE WORK
A Bug report helps the developers in identifying the bugs and resolving them. But often misclassification is an issue that sacrifiscetheaccuracyofbugreports. So in this paper we have 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 accuracy rates as 92.02% based on confusion matrix. We have trained the supervised bug report, for future work we suggest to implement the same methodology of bug report classification and optimization on unsupervised bug report.

REFERENCES
1. Ashima Kukkara, Rajini Mohana: A Supervised Bug Report Classification with Incorporate and Textualfield Knowledge. International Conference on Computational Intelligence and Data Science (ICCIDS2018)


AUTHORS PROFILE

Gayathri P M MCA Scholar at Amrita Vishwa Vidyapeetham University, Amrita School of Arts & Sciences, Kochi, Kerala, India.

Greeshma KBabu MCA Scholar at Amrita Vishwa Vidyapeetham University, Amrita School of Arts & Sciences, Kochi, Kerala, India.

Deepa G Assistant Professor at Amrita Vishwa Vidyapeetham University, Amrita School of Arts & Sciences, Kochi, Kerala, India.