

# Bug Triage : An Automated System of Bug Reduction

Prof. Pooja Bhagat -Thakre<sup>1</sup>, Prof. Vishal Wagh<sup>2</sup>, Prof. Rachana Mudholkar<sup>3</sup>, Prof. Vishal Patil<sup>4</sup>  
*Department of Computer Engineering, Dr. D.Y. Patil Institute of Engineering, Management and Research, Pune, India*

*Email:pooja.bhagat@dypiemr.ac.in<sup>1</sup>,vish.wagh143@gmail.com<sup>2</sup>,mudholkar.rrm@gmail.com<sup>3</sup>,patilvd85@gmail.com<sup>4</sup>*

**Abstract-**Bug removal is a fundamental step during bug fixing. Bug removal is the way toward fixing bug whose primary target is to accurately apportion a designer to another bug additionally taking care of. Many software organizations spend their majority of cost in managing these bugs. To reduce the time cost in manual work and to improve the working of programmed bug removal, two procedures are connected in particular content characterization and double arrangement. The component choice calculations to at the same time reduce the information scale and upgrade the correctness of the bug reports in the bug removal. The important thing is the need to build up a powerful model for doing information lessening on bug information set which will decrease the size of the information and increment the nature of the information., by minimizing the time and cost. System produces bug report and assigns that bug to appropriate developer.

**KEYWORDS:** Bug fixing, data reduction, Instance selection, Feature selection, Data Mining, Clustering

## 1. INTRODUCTION

Many software companies spend most of the money in fixing the bugs. Large software projects have bug repository that collects all the information related to bugs. In bug repository, each software bug has a bug report. The bug report consists of textual information regarding the bug and updates related to status of bug fixing [1]. Once a bug report is formed, a human error assigns this bug to a developer, who will try to fix this bug. This developer is recorded in an item assigned-to. The assigned to will change to another developer if the previously assigned developer cannot fix this bug. The process of assigning a correct developer for fixing the bug is called bug triage [2]. Bug removal is one of the most time consuming step in handling of bugs in software projects. Manual bug removal by a human is time consuming and error-prone since the number of daily bugs is large and lack of knowledge in developers about all bugs. Because of all these things, bug triage results in expensive time loss, high cost and low accuracy [3]. The information stored in bug reports has two main challenges. Firstly the large scale data and secondly low quality of data. Due to large number of daily reported bugs, the number of bug reports is scaling up in the repository. bugs are degrading the quality of bug reports redundantly. The effective bug triage system is proposed which will reduce the bug data to save the labor cost of developers. It also aims to build a high quality set of bug data by removing the redundant and non-informative bug reports [4]. Putting away points of interest of bugs, it assumes a critical part. Bug stores are broadly utilized for keeping up programming bugs, e.g.,

a well-known and open source bug store, Bugzilla. A recorded bug is known as a bug data, once a bug report

is created, a human error removal doles out this bug to a designer, who will endeavor to settle this bug [2]. A designer, who is allocated to another bug report, begins to settle the bug in view of the information of chronicled bug settling. Bug removal is a routine of passing the settling bugs to correct engineer. An engineer, who is relegated to an imaginative bug data, begins to settle the bug based on the data of past bug settling. Regularly, the engineer pays endeavors to perceive the new bug report and generally settled bugs as a kind of perspective (e.g., hunting down comparative bugs and applying available answers for the new bug) [3]. Status of a bug report is changed by late consequence of taking care of this bug until the bug is totally settled. Displayed work utilizes the methodologies in view of content demeanor to help bug removal. In such a way, the rundown of a bug report are concentrate as the printed content while the engineer who7 can join this bug is as the name for grouping. It gives low precision [4].

## 2. PROPOSED SYSTEM DESIGN

### 1. Objectives:

- Propose a combination approach to addressing the problem of data reduction
- Removal Bug to developer which having most experience on that problem.
- To generated Bug report with the bug suggestion.

### Problem Statement:

Bug removal is an essential step in the process of bug fixing. Bug removal is the process of fixing bug whose main objective is to correctly allocate a developer to a new bug for further handling by using two techniques are applied namely Text Classification algorithm and Binary classification algorithm. The output of system is bug allocate to appropriate developer which having more skill to solve that bug.

### 3. PROJECT OVERVIEW:

The goal of bug triage is to assign a bug to the correct potential developer. In bug triage, a bug data set is converted into a text matrix with two proportions, namely the bug dimensions and word dimension. Bug reports are in the form of summary and description. Bug data decrease to compress the scale and to improve the quality of data in bug repositories which is applied as a stage in data preparation of bug removal. Existing techniques of instance selection and feature selection are used to remove definite bug reports and words [4]. Classifiers are also used to fix new bug report. It also used to predicate developer to fix bug. It replace the original data set with the reduced data set for bug triage. Instance selection and feature selection are widely used techniques in data processing. Due to the large number of daily bugs and the lack of expertise of all the bugs, manual bug removal is expensive in time cost and low in accuracy. To avoid the expensive cost of manual bug removal, existing work has proposed an automatic bug removal approach, which applies text classification techniques to predict developers for bug reports [5]. In this approach, a bug report is mapped to a document and are late developer is mapped to the label of the document. Then, bug removal is converted into a problem of text classification and is automatically solved with mature text classification techniques.

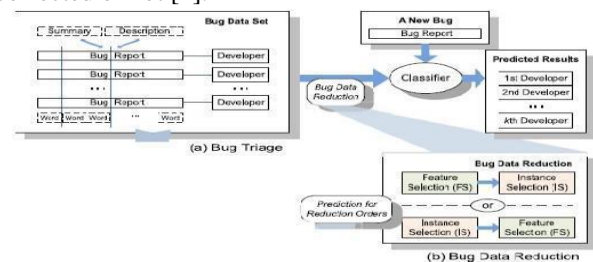
### 4. DEVELOPMENT METHODOLOGY

The single problem can be solved by different solutions. This considers the performance parameters for each approach. Thus considers the efficiency issues. The system will have a database containing files along with encryption and decryption keys. The end user will access these files and the owner will upload the files with the key which only the authorized user can access to download the file. This solves the problem of unauthorized access. Unravel completely different components of the matter (sub problems), then mix the resolutions of the sub issues to succeed in Associate in Nursing overall solution. Usually once employing an additional naive technique, several of the sub issues are generated and solved over and over [6]. The dynamic programming approach seeks to unravel every sub downside just once, so reducing the quantity of computations: once the answer to a given sub downside

has been computed, it's stored; consequent time an equivalent resolution is required, it's merely researched. This approach is particularly helpful once the quantity of continuation sub issues grows exponentially operate of the dimensions of the input. This system to have a tendency to use this technique for authentication purpose. The user must register 1st and also the details are hold on within the information. The registered user are hold on along with his identity, whenever the authority must certify he can merely consider information for the desired details [7].

#### 4.1 System Architecture

Figure shows the architecture of the bug triaging. Aim of bug triage is to allocate a developer for bug deception. Once a developer is allocated to a new bug report they will solve the bug or attempt to fix it. They will provide the status associated to bug whether it is corrected or not [1].



#### A. Interactions with Bug Reports

People play different roles as they interact with reports in a bug repository. The person who submits the report is the reporter or the submitter of the report. The person who decides if the report is meaningful and who assigns responsibility of the report to a developer. The one that resolves the report is the resolver. A person that configure 2: A sample Bugzilla bug report from Eclipse. Table 1: Daily bug submissions around and after product release. Around Release After Release Mean Min Max Mean Min Max Eclipse 48 1 192 13 1 124 Firefox 8 1 37 5 1 37 tributes a fix for a bug is called a contributor. A contributor may also contribute comments about how to resolve a bug or additional information that leads to the resolution of a report. A person may assume any one of these roles at any time. For example, a removal may resolve a report as the duplicate of an existing report. Alternatively, a developer may submit a report, assign it to himself, contribute a fix, and then resolve the report. For that report, a single person has fulfilled all the roles.

#### 4.2. Bug Triage

The goal of bug triage is to assign a new-coming bug to the correct potential developer. In bug triage, a bug data set is converted into a text matrix with two dimensions, namely the bug dimensions and word dimension.

#### 4.3. Graph Module

This module show's four part's as follow:

- 1) Firstly it will show how many bugs are not assigned to any developer. It will give complete status about the bugs to the admin so that he will come to know which bugs are not assigned yet.
- 2) Secondly it will show how many bugs are not assigned to any developer. It will give complete status about the bugs to the admin so that he will come to know which bugs are assigned.
- 3) Thirdly it will show how many bugs are rectified by the developer's. It will give complete status about the bugs to the admin so that he will come to know which bugs are rectified completely.
- 4) Fourthly it will show how many bugs are not rectified by the developer's. It will give complete status about the bugs to the admin so that he will come to know which bugs are not rectified yet.

Figure 1: A sample Bugzilla bug report from Eclipse.

**Bugzilla Bug 4746** DOR: TreeItem needs removeAll() method (GGONLJ) Last modified: 2005-04-06 11:01:35

Bug List (This bug is not in your last search results) [Show last search results](#) [Search rate](#) [Enter new bug](#)

**Bug 4746**  
 Bug:   
 Product: Platform  
 Component: SWT  
 Status: RESOLVED  
 Resolution: FIXED  
 Assigned To:   
 QA Contact:   
 URL:   
 Summary: DOR: TreeItem needs removeAll() method (GGONLJ)  
 Status:   
 Keywords:

Hardware: All  
 OS: All  
 Version: 2.0  
 Priority: P1  
 Severity: normal  
 Target Milestone:

Reporter:   
 Add CC:   
 CC:

Attachment	Type	Created	Size	Actions
<a href="#">Create a New Attachment</a> (proposed patch, testcase, etc.)				<a href="#">View All</a>

Bug 4746 depends on:   
 Bug 4746 blocks:  [Show dependency tree](#)

Votes: 0 [Show votes for this bug](#) [Vote for this bug](#)

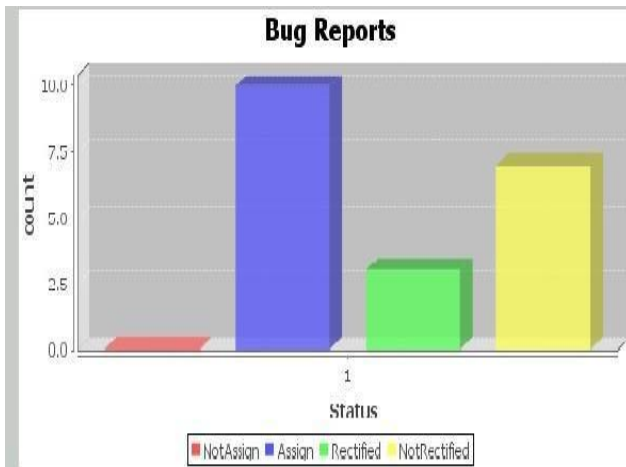
Description: [\[read\]](#) Opened: 2001-10-11 14:22

TreeItem should really have a removeAll() method just like Tree.

NOTES:

----- Comment #1 Proulx 2007-10-20 16:33 (read) -----

PRODUCT VERSION:  
 Build 125



## REFERENCES

- [1] J. Anvik, L. Hiew, and G. Murphy, "Who should fix this bug?" in Proc 28th International Conference on Software Engineering. ACM, 2006, pp. 361–370.
- [2] Anvik, J. 2007. Assisting Bug Report Triage through Recommendation. Ph.D. Dissertation, University of British Columbia
- [3] G. Jeong, S. Kim, and T. Zimmermann, "Improving Bug Triage with Tossing Graphs," Proc. 17<sup>th</sup> ACM SIGSOFT Symp. Foundations of Software Engineering (FSE '09), Aug. 2009, pp. 111-120.
- [4] Herraiz, D. German, J. Gonzalez-Barahona, and G. Robles, "Towards a Simplification of the Bug Report Form in Eclips," in 5th International Working Conference on Mining Software Repositories, May 2008.
- [5] Jonnakuti Ramesh 1 M.Tech II year" An Automated Approach or Software Bug classification "[http://ijesat.org/volumes/2013\\_Vol\\_03\\_Iss\\_025/IJSEAT\\_2013-03\\_05\\_12](http://ijesat.org/volumes/2013_Vol_03_Iss_025/IJSEAT_2013-03_05_12).
- [6] A. Hotho, A. Nürnberger and G. Paaß, "A Brief Survey of Text Mining," vol. 20, GLDV Journal for Computational Linguistics and Language Technology, 2005, pp. 19-62.
- [7] A. E. Hassan, "The Road Ahead for Mining Software Repositories," IEEE Computer society, pp. 48-57, 2008.
- [8] S. Diehl, H. C. Gall and A. E. Hassan, "Special issue on mining software repositories," in Empirical Software Engineering An International Journal © Springer Science+Business Media, 2009.
- [9] L. Yu, C. Kong, L. Xu, J. Zhao and H. Zhang, "Mining Bug Classifier and Debug Strategy Association Rules for Web-Based Applications," in 08 Proceedings of the 4th international conference on Advanced Data Mining and Applications, 2008.
- [10] T. Menzies and A. Marcus, "Automated severity assessment of software defect reports," in IEEE International Conference on Software Maintenance, 28 2008-Oct. 4 2008, pp. 346–355.