ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

Potent Bug Assortment Using Data Reduction Techniques

Karishma Musale , Gorakshanath Gagare

A M.E Student, Department of Computer Engineering, SVIT, Nasik, India

Assistant Professor, Department of Computer Engineering, SVIT, Nasik, India

ABSTRACT: Companies of software pays lot of money for fixing bugs. If the system has bugs, nothing will work properly in it. So, It is necessary to solve the bugs in software. Here, we address the data reduction for bug triage i.e. to increase the quality of bug data by reducing the bug dimension. Also we are using instance selection and feature selection method for extracting attributes from new bug dataset. Mozilla and eclipse are the available dataset for it. Also after data reduction. We are showing the best fit solution for required bug issue.

KEYWORDS: Instance selection, Feature selection, data reduction.

I. INTRODUCTION

In software engineering practices, identification and correction of bugs are main task. For example, per day lot of bugs are reported in Eclipse and Mozilla. Quality of dataset is loss because of bugs. After finding bugs in software, bugs are given to developers for fixing. Assigning bugs to developers is very difficult and time consuming task .Bug tracking system is used for most of big software projects for tracking/managing bugs. Companies spend lot of money on fixing bugs. After getting solutions from solver, tester, developers , save this data in repository which is called as bug repository. This bug report contains the summary and description of the bugs. Summary is used to identify the bugs. Description gives the solution of bugs. We identify bugs using their summary. Our main task is not only reducing the bug dataset but also improving the quality of bug dataset. We use instance selection and feature selection methods for reducing the bug dataset. When we use instance selection, accuracy gets decreased. So, feature selection is used. We are using both techniques combinely. Developers store their solutions in bug repository. In previous system can not get best solution. So, we are showing the best fit solution among all solutions.

Objectives of proposed work is as follows:

- Simultaneously decrease the scales of word dimension and bug dimension.
- Improving the bug fixing.
- Improve bug fixing results of reducing data and gives solution with specific task.

II. RELATED WORK

D. Cubranic and G. C. Murphy [2] used an application of supervised machine learning using a naive Bayes classifier for automatically assign bug reports to developers. For that they experimented their approach on bug reports from a large open-source project such as Eclipse.org. And get 30 percent classification accuracy. S. Kim, K. Pan, E. J. Whitehead, Jr [7] used project-specific bug finding tool using memories of bug fixes. They used BugMem tool to detect potential bugs which suggests corresponding fixes. They tackle information to improve the quality of source code and provide detailed guidance to developers. G. Jeong , S. Kim, and T. Zimmermann [3], studied for improving the bug assignment process and reduce unnecessary tossing steps, they used tossing graph model which used existing tossing history. This system assigns bug to expert developers. It gets 23 % accuracy as compared to existing system. J. W. Park, M. W. Lee, J. Kim, S. W. Hwang, and S. Kim, [4], used COSTRIAGE (A Cost-Aware Triage Algorithm for Bug Reporting Systems) technique. They used a proof-of-concept implementation by using cost of bug fixing time. This technique is used to minimize the cost of bug fixing. C. Sun, D. Lo, S. C. Khoo, and J. Jiang [8], improved the

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

accuracy of duplicate bug retrieval. In order to identify such duplicates accurately, they proposed a retrieval function (REP) to measure the similarity between two bug reports. For more accurate measurement of textual similarity, they extend BM25F technique for duplicate report retrieval.

III. PROPOSED SYSTEM

This architecture gives the details of proposed system. Here, we are using available bug dataset as eclipse and Mozilla. After taking the bug dataset , this dataset is classified . We are using instance selection and feature selection algorithms for reduction of dataset as well as improving the quality of bug dataset. After getting reduced data, shows the best fit solution.

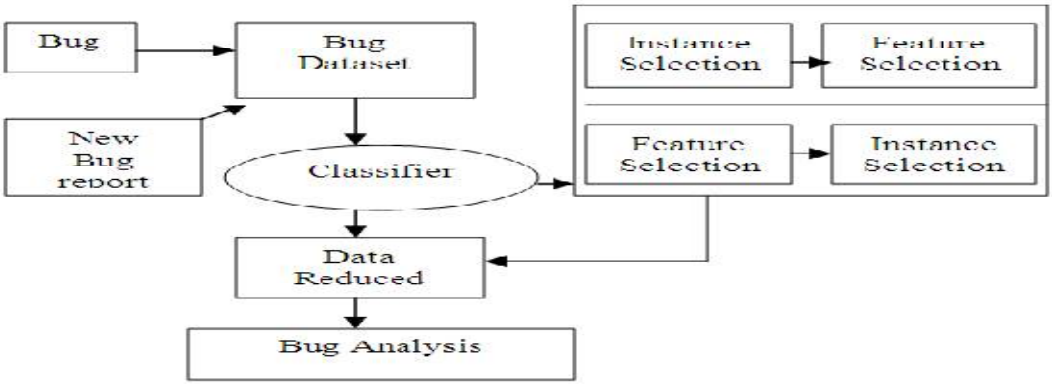

Fig 1.System Architecture

IV. RESULTS

Following gives the result of proposed system :

1) The **Importing Dataset**: This shows the dataset upload while clicking on import button.

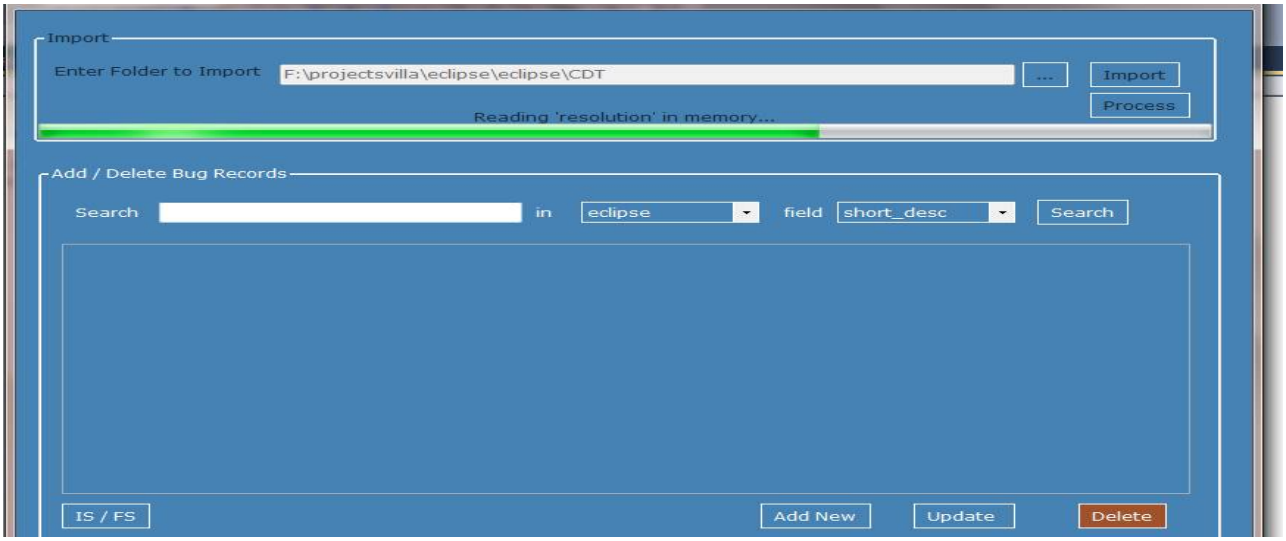

Fig 2. Importig Dataset

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

2) **Reduction Of Dataset**:This shows the reduction of dataset in proposed system with existing system dataset and display number of records reduced in proposed system.

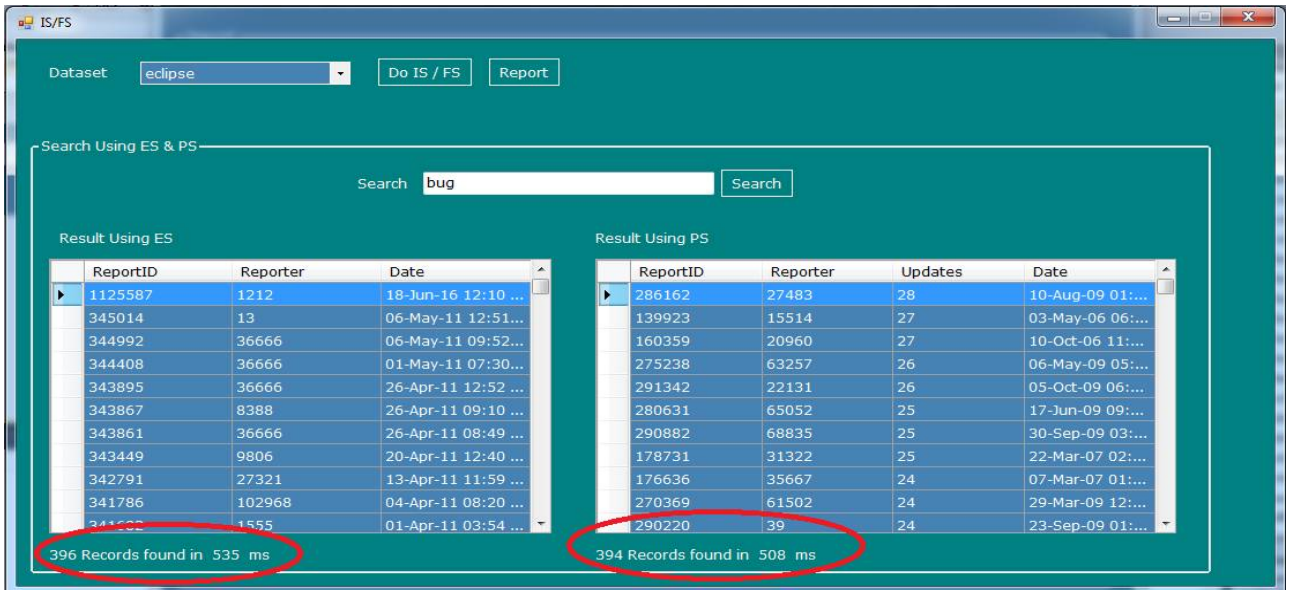

Fig 3. Data Reduction

3) **Removing Duplicate Records**: This shows the duplicate records which are deleted for data reduction.

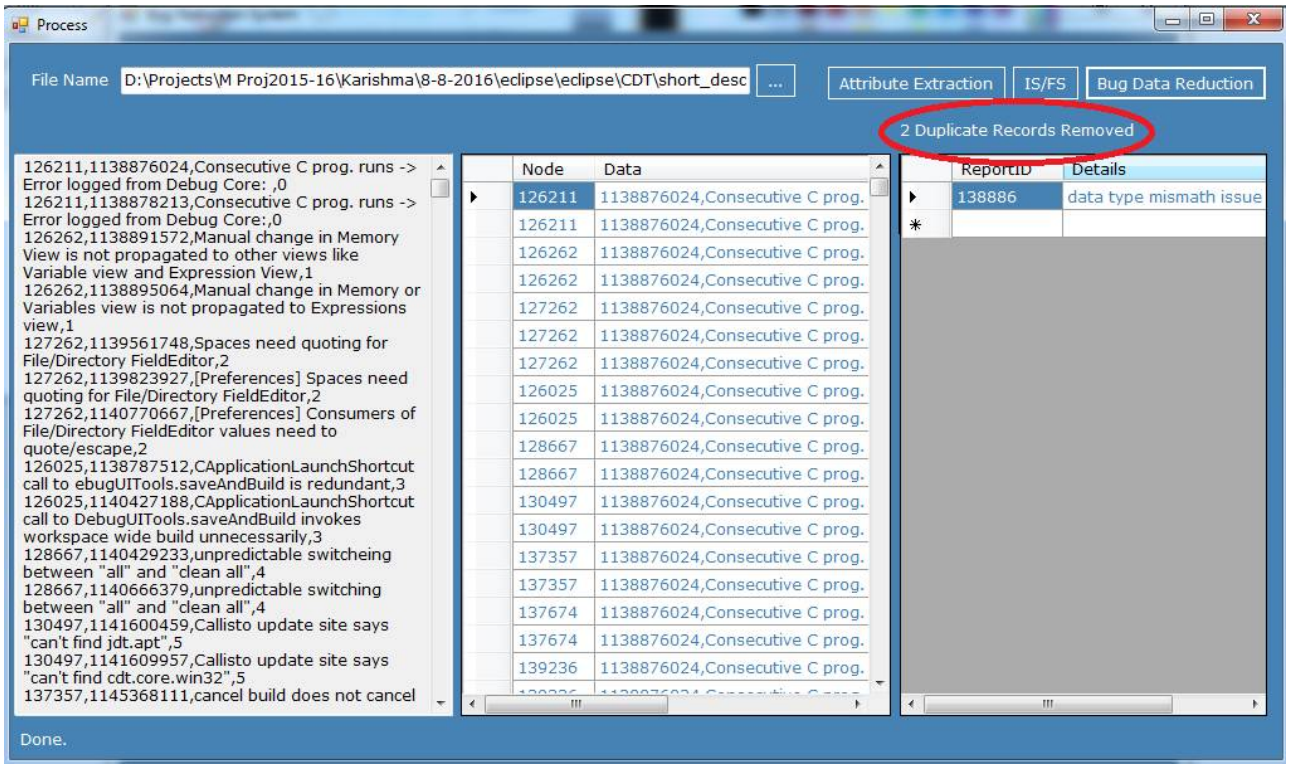

Fig 4. Removing duplicate records

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 9, September 2016

V. CONCLUSION AND FUTURE WORK

The Software Companies spend lot of on fixing bug. Fixing bug is very important. Time required for this is very long. So, we are implementing this system for reducing cost, reducing bug dataset and improving the quality of bug dataset. We use Instance selection and feature selection both techniques combines to achieve data scaling and quality of bug dataset. And additionally search best fit solution among all solutions. So time is saving as well as space to store bug dataset is minimize. In future work, we plan on improving the data reduction based on their attribute values.

REFERENCES

1. B Jifeng Xuan, He Jiang, Yan Hu, Zhilei Ren, Weiqin Zou, Zhongxuan Luo, and Xindong Wu, 'Towards Effective Bug Triage with Software Data Reduction Techniques', ieee transactions on knowledge and data engineering. vol.27, no. 1, january 2015.
2. D. Cubranic and G. C. Murphy, 'Automatic bug triage using text categorization', in Proc. 16th Int,Conf. Softw. Eng,Knowl. Eng.,Jun. 2004, pp. 9297.
3. S. Kim, K. Pan, E. J. Whitehead, Jr., 'Memories of bug fixes' in Proc. ACM SIGSOFT Int.,Symp. Found. Softw. Eng.,2006, pp. 3545
4. Jeong, S. Kim, and T. Zimmermann, 'Improving bug triage with tossing graphs', in Proc. Joint Meeting 12th Eur.Softw. Eng. Conf. 17th ACMSIGSOFT Symp. Found. Softw.Eng.,Aug. 2009,pp. 111120.
5 J. W. Park, M. W. Lee, J. Kim, S. W. Hwang, and S. Kim, 'Costriage: A cost-aware triage algorithm for bug reporting systems', in Proc. 25th Conf. Artif. Intell., Aug. 2011, pp.139144.
6 A. E. Hassan, 'The road ahead for mining software repositories', in Proc. Front. Softw. Maintenance, Sep. 2008 pp. 4857.
7 J. Xuan, H. Jiang, Z. Ren, and W. Zou, 'Developer prioritization in bug repositories', in Proc. 34th Int. Conf. Softw. Eng., 2012, pp. 2535.
8. C. Sun, D. Lo, S. C. Khoo, and J. Jiang, 'Towards more accurateretrieval of duplicate bug reports', in Proc. 26th IEEE/ACM Int. Conf. Automated Softw. Eng., 2011, pp. 253262.
9. H. Brighton and C. Mellish, 'Advances in instance selection for instance- based learning algorithms', Data Mining Knowl. Discovery. vol. 6, no. 2, pp. 153172, Apr. 2002.
10. C. Sun, D. Lo, S. C. Khoo, and J. Jiang, 'Towards more accurate retrieval of duplicate bug reports', in Proc. 26th IEEE/ACM Int. Conf. Automated Softw. Eng., 2011, pp.253262.
11. H. Brighton and C. Mellish , 'Advances in instance selection for instance-based learning algorithms', Data Mining Knowl . Discovery, vol. 6, no. 2, pp. 153172, Apr. 2002.
12. Y. Yang and J. Pedersen,' A comparative study on feature selection in text categorization' , in Proc. Int. Conf. Mach. Learn., 1997, pp. 412420.
13. S. Artzi, A. Kiezun, J. Dolby, F. Tip, D. Dig, A. Paradkar, and M. D. Ernst, 'Finding bugs in web applications using dynamic test generation and explicit-state model checking' , IEEE Softw. vol. 36 no. 4, pp. 474494, Jul./Aug. 2010.
14. Y. Fu, X. Zhu, and B. Li, A survey on instance selection for active learning, Knowl. Inform. Syst., vol. 35, no. 2, pp. 249283, 2013.
15. M. Grochowski and N. Jankowski,' Comparison of instance selection algorithms ii, results and comments ', , in Proc. 7th Int. Conf. Artif. Intell. Softw. Comput., Jun. 2004, pp. 580585.
16. T. M. Khoshgoftaar , K. Gao , and N. Seliya, ' Attribute selection and imbalanced data: Problems in software defect prediction', in Proc. 22nd IEEE Int. Conf. Tools Artif. Intell., Oct. 2010, pp. 137144.
17. S. Kim, H. Zhang, R. Wu, and L. Gong, 'Dealing with noise in defect prediction', in Proc. 32nd ACM/IEEE Int. Conf. Softw. Eng., May 2010, pp. 481490.
18.] A. Lamkanfi, S. Demeyer, E. Giger, and B. Goethals, 'Predicting the severity of a reported bug ', in Proc. 7th IEEE Working Conf. Mining Softw. Repositories, May 2010, pp.110.
19. E. Murphy-Hill, T. Zimmermann, C. Bird, and N. Nagap- pan, 'The design of bug fixes', in Proc. Int. Conf. Softw. Eng.,2013, pp. 332 341.

BIOGRAPHY

Karishma C. Musale received the B.E. degrees in Information Technology from K.B.Thakare College of Engineering, Nashik, Savitribai Phule Pune University in 2014. Now pursuing M.E. in Computer Engineering from Sir Visvesvaraya Institute of Technology,Chincholi, Nashik, India.

Prof. Gorakshanath J. Gagare received M.Tech (CSE) from BVUCOE, Pune in 2012. Currently working as Assistant Professor of Computer Engineering Department in Sir Visvesvaraya Institute of Technology, Chincholi, Nashik, India.