Advances in Parallel Computing Algorithms, Tools and Paradigms
D.J. Hemanth et al. (Eds.)

doi:10.3233/APC220025

Automatic Bug Tracking System Using Text Analysis and Machine Learning Predictions

DR. P Meena Kumari[1]
[1]*Associate Professor, T John Institute of Technology, Bengaluru, Karnataka,*

Abstract. Every day, many bugs are raised, which are not fully resolved, and a large number of developers are using open sources or third-party resources, which leads to security issues. Bug-triage is the upcoming automated bug report system to assign respective security teams for an ample rate of bug reports submitted from different IDEs within the organization (on-premises). Furthermore, by predicting the appropriate team (who can resolve the bug) in an organization, the bugs can be assigned once it is tracked. With this, cost and time can be saved in tracking and assigning the bugs. In this paper, we are implementing an Automatic bug tracking system (ABTS) to assign the team for the reported bug using the Text analysis for bug labeling and classification machine learning algorithm for predicting developer.

Keywords. Supervised Machine Learning, Text Analysis, Bug Triage, Natural Language Processing

1. Introduction

For the past coupleof decades, self-service software has been created using machine-learning techniques. In addition, to evaluate patterns in various applications SML (Supervised Machine Learning) methods are widely used, but we have found little for software repositories.

While developing open-source software, Bug tracking systems play an essential role as team members can be dispersed worldwide. Usually,the bug report is assigned manually to a single developer, who is then responsible for correcting thoseerrors. In such widely distributed projects, developers and other project contributors can never meet, where there are many integrity and confidentiality issues. Therefore, we are implementing the Automatic Bug Tracking System (ABTS) for the reported bug description text. Once in the ABTS a bug is reported then it will be stored in the database in the form of .txt, CSV, or excel files, it is further analyzed and processed to assign and analyze its features and fix them.

Automated text classification is considered an essential method for organizing and processing many documents in digital forms, which are becoming more and more widespread. In general, text classification plays an essential role in capturing information, retrieving text, and answering questions. In addition, finding and fixing bugs in software systems has always been a significant issue in software engineering.

[1]Dr. P Meena Kumari, Associate Professor, T John Institute of Technology, Bengaluru, Karnataka; E-mail: meenasri33@gmail.com

2. Problem Statement

In the existing system, bug reporting and tracking are manual, that is, when an application is implemented in an organization; if ever bug reported in that IDE, the testers keep them largely available open sources like google, etc., to security issues. Therefore, the newly implementing application details may leak outside easily, even though the bugs are not solved. In this manual system, many issues are there like security, integrity, and technical.

3. Literature Survey

Chaturvedi & Singh, 2012, stated that to send or report errors and monitor their progress various bug reporting systems had been developed. Bug reporting and tracking systems provide a platform for recording problems/failures encountered by the client or software user [1].

As per Zou et al. (2011), current approaches to bug classification are based on machine learning algorithms that generate classifiers from bug reporting training[2]. Bug sorting is an important step in the bug correction process. The goal of bug sorting is to assign a new incoming bug to the right potential developer.

Anvik et al. (2005)state that most open-source software development projects include an open bug repository - which software users can take full advantage of - to report and track software system issues and the scope for improvement is high. Using the Open Bug Store: more system problems can be detected due to the relative ease of reporting errors[3]. In addition, more problems can be solved because more developers can participate in troubleshooting, and developers and users can participate in bug discussions and allow users to enter the system direction. Nevertheless, many integrity and confidentiality issues arise for many newly implemented applications.

Čubranić, (2004) stated that Bug Triage, which decides what to do with the incoming bug report, is increasingly taking resources for developers on large open-source projects[4]. According to T et al. (2019), text classification is becoming an essential step in capturing data and adequately categorizing websites [5].Neethu & Rajasree, (2013)stated that, unfortunately[6], many things have diverted users faster than the notion that developers are unresponsive and ignore user bug messages and feature requests - and kill the project community.

According to Luyckx & Daelemans, (2005), Automatic text classification is a text mining application that refers to documents by pre-defined content categories[7]. Many text classification applications include indexing, document filtering or routing, and hierarchical classification of websites and web search engines.

Pushaplatha and Murnalini 2019 have proposed a severity prediction and bug report classification for closed software bugs [8]. In this approach, three different ensemble ML algorithms such as Bagging, voting, and Adaboosthave appliedto NASA dataset. Ramay et al., 2019 [12] have proposed deep neural network-based bug severity prediction techniques[9]. At first, the NLP methods are utilized to preprocess the bug's reports and afterward another score is determined to track down the feeling of bug columnists. Otoom et al., 2019 have introduced a mechanized methodology for bug reports arrangement utilizing a clever component determination strategy [10]. The list of capabilities is created based on the normal events of the various watchwords in the rundown of the bug report.

4. Methodology

The process of converting structured text data into meaningful data for analysis, text classification, supporting sentimental search tools analysis to measure product review feedback from customers, and entity modeling for fact-based decision making is called Text analytics. **Figure 1**, Text analysis is the classification of specific text according to several author categories. An element vector is created in a document that contains binary or numeric characters separated by commas based on the information and class status (A and B).

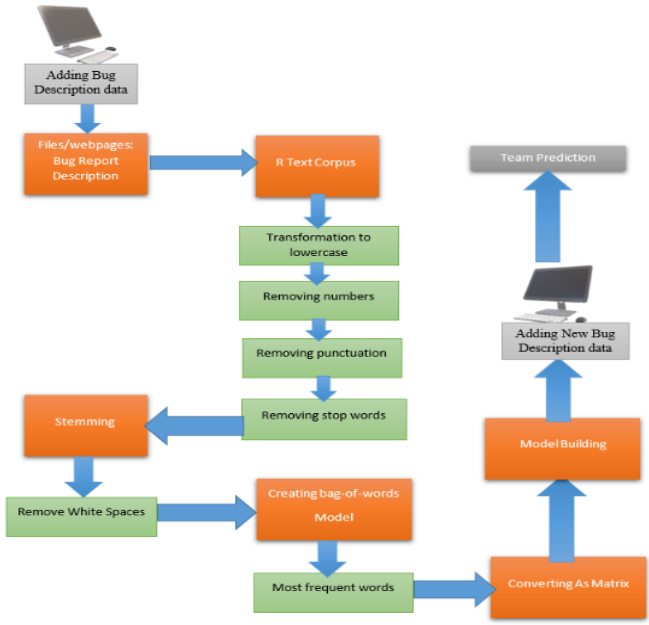


Figure 1. Text analyzer Architecture

Classification algorithms take the identified data (because they are in the monitored practice methods) and learn the patterns in the data that can be used to estimate the classification output variable(*Machine Learning*, n.d.). It can most often be group variable (a variable that determines which group a particular case belongs to) and binomial (two groups) or multinomial (more than two groups), and these problems are quite common ML tasks.

Estimating the class of a given data point is the process of Classification. Classes are sometimes called targets or labels or categories. Classification of Predictive Modeling is the task of estimating the mapping function (f) from input variables (x) to discrete output variables (y). The problem of identifying subpopulations (categories) to which it belongs to based on the target label is called Classification. In taxonomy there are many applications in different domains, like credit approval, medical diagnostics, target marketing, and so on. After selecting and modifying the elements, the documents can be easily referenced in a usable form using the ML algorithm. However, they often

differ in the Random Forest Model, Naïve Bayes, and XGBoost ML (Machine Learning) algorithms.

Naive Bayes (NB) is a very simple probabilistic algorithm based on Bayes theorem which is used in classification problem solving. In Bayesian analysis, the resulting taxonomy is created by combining both information categories, the former and the probabilities, to create the so-called posterior probability of the Bayes' rule. The NB classification is a primary probability classification based on the application of the Bayes theory. NB calculates the set of probabilities by combining the values in a given dataset. The NB classification also has a fast-decision-making process.

Random Forest creates more decisive trees and combines them to get a more accurate and stable reference. RF (Random Forest) is a flexible and easy-to-use ML algorithm that gives great results most of the time, even without hyper-parameter tuning. Random Forest creates more decisive trees and combines them to get a more accurate and consistent reference. **Figure 2**, Random Forest adds extra coincidence to the model when the tree grows. Instead of looking for the most important function when splitting a node, it looks for the best function among the random subset functions. This means that the data/population used to build the random forest tree will be replaced, and the detailed variables will also be bootstrapped, so the partitioning will not be implemented on the same important variables.

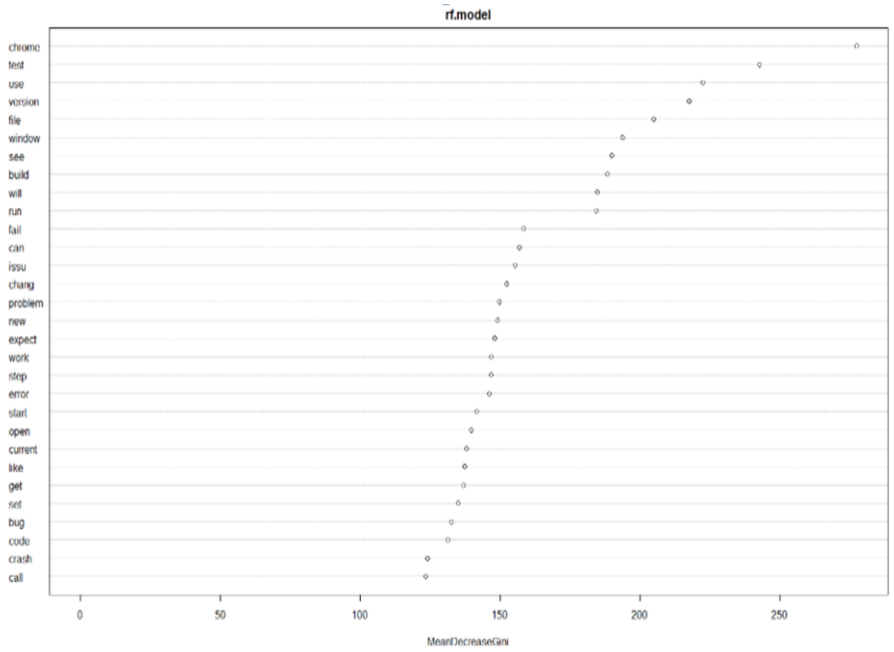


Figure 2. Most frequent words

Implementing the gradient boosting concept is *XGBoost*. Algorithm writer Tianki Chen, says that “To give better performance XGBoost uses a more controlled model formalization to control over-fitting”. Therefore, it helps in reducing excess equipment. It is built on the principles of the gradient improvement framework and is designed to "change the intensities of machine computing constraints and provide a scalable, portable and accurate library." To calculate the best distribution XGBoost uses

histogram-based and a pre-computed algorithm. The histogram-based algorithm divides all element data points into separate baskets and uses these baskets to find the histogram distribution. XGBoost not only performs taxonomic functions; Accepts only numeric values. Therefore, different encodings like one-time encoding, label encoding or intermediate encoding, must be performed before classification data can be delivered to XGBoost. XGBoost can make automatic function selection and capture high-order interactions without hurting. XGBoost also has a randomization parameter, i.e., a partial pattern of columns, which helps to reduce the interaction of each tree. It can deal with persistent and categorical data naturally, it can deal with naturally missing data, and they are strong for outliers at the inputs, which vary under the constant changes of the inputs. Select the default variable, and we can capture linear relationships in data and high-order interactions between inputs and scale large datasets.

5. Analysis, Evaluation, and Results

In this article, we have collected publicly available bug-reported datasets in the CSV file format, and the description of the bug is taken as an input text variable. For this text variable, we have analyzed and by applying effectively *Supervised- ML Classification Algorithms* in *R-Studio* environment using *R-Programming,* to predict the team which the ABTS can automatically assign that newly reported bug Furthermore, from the respective supervised ML classification algorithms, the "Accuracy" has been taken as evolution metrics to determine the effective machine learning algorithm for the input database. **Figure 3**, Input database is a Bug reported data with four variables ("owner," "issue_title," "description", and "Team" (A &B)) and 5000 observations. Here we are taking the "description" variable for the text analysis and applying *classification* algorithms, as the dependent (target/outcome) variable has two labels, which are considered binary classifications.

	owner	issue_title	description	Teams
1	amit@chromium.org	Scrolling with some scro...	Product Version : <see aboutversion>URLs (if applicable) :0.2.149.27Other browsers tested: Firefox / IEAdd OK or FAIL after other browsers where you...	A
2	jon@chromium.org	Proxy causes some or all...	Product Version : 0.2.149.27 (1583)URLs (if applicable) : http://www.igoogle.com, http://code.google.com/p/chromiumOther browsers tested:Add OK ...	A
3	pfeldman@chromi...	Web inspector button "d...	Product Version : chrome beta 1URLs (if applicable) :Other browsers tested:Add OK or FAIL after other browsers where you have tested this issue: S...	B
4	jon@chromium.org	Habari admin interface i...	Product Version : 0.2.149.27 (1583)URLs (if applicable) :Other browsers tested:Add OK or FAIL after other browsers where you have tested this issue: ...	A
5	pkasting@chromiu...	Maximize on second lar...	Product Version : 0.2.149.27URLs (if applicable) :What steps will reproduce the problem?1. Open Chrome on the primary monitor which has a smaller ...	B
6	jon@chromium.org	Adding music in Pitchfor...	Product Version : 0.2.149.27URLs (if applicable) :Other browsers tested: Firefox3, Opera9, IE7Add OK or FAIL after other browsers where you have test...	A
7	brettw@chromium...	Attempting to open Chr...	Product Version : Windows XP 64-bitURLs (if applicable) :Other browsers tested:Add OK or FAIL after other browsers where you have tested this issue:...	A
8	jon@chromium.org	Constant "Unknown erro...	Product Version : 0.2.149.27URLs (if applicable) : http://news.bbc.co.uk/1/hi/technology/default.stm, http://forum.preys-world.com/index.php and ma...	A
9	amit@chromium.org	Scrolling with some scro...	Product Version : <see aboutversion>URLs (if applicable) :0.2.149.27Other browsers tested: Firefox / IEAdd OK or FAIL after other browsers where you...	A
10	darin@chromium...	MemcheckCond error in...	This is introduced between 53415 - 53406 (http://build.chromium.org/buildbot/memory/builders/Chromium%20Mac%20(valgrind)/builds/6676)I doubt i...	A

Figure 3. Input data in table format

Creating corpus, This Vignette ***tm*** package briefly introduces text mining in R using the text mining framework provided. We demonstrate methods for importing data, corpus processing, preprocessing, metadata management, and creating a template for term documents. **Figure 4**, The main structure for document management in the dark is the so-called corpus, which refers to collecting text documents. The corpus is an abstract concept and may have many implementations in parallel. The so-called default

execution is VCorpus (short for unstable corpus), which executes known meanings from most R objects: corpora means that R objects are full memory.

Input Text:

```
[1] "Product Version      : <see about:version>URLs (if applicable)
:0.2.149.27Other browsers tested: Firefox / IEAdd OK or FAIL after other
browsers where you have tested this issue:Safari 3:    Firefox 3: OK         IE
7:OKWhat steps will reproduce the problem?1. Open any webpage on compaq 6715s
running vista.2. Try scrolling with the touchpad3. Scrolling down will work ,
but up will not.What is the expected result?The page to scroll up.What happens
instead?The page doesn't move.Please provide any additional information below.
Attach a screenshot if possible.Only a minor bug. "
```

Figure 4: Corpus data for the description variable

6. Transformations

Once we have the corpus, we usually want to modify the documents, such as ignoring tracks and deleting tracks. In tm, this whole function is included in the transition concept. **Figure 5**, Transitions are performed using the tm_map() function, which applies the function (maps) to all corpus elements. Generally, all transitions work on text documents and only apply to all the tm_map () corpus.

6.1 Creating dtm (Bag-Of-Words) Model:

Creating Term-Documents Matrices: A common approach to text extraction is to create a matrix of term documents from the corpus. **Figure 6**, The TermDocumentMatrix and DocumentTermMatrix classes (depending on whether you want the expressions to be rows and columns as documents) use smaller matrices for corpora in the Tm package. Word-document matrix checking displays a pattern, while as. Matrix () presents the entire matrix in a dense format (which is very memory-intensive for large matrices).

<<DocumentTermMatrix (documents: 109982, terms: 598023)>>
Non-/sparse entries: 4002301/65767763285
Sparsity : 100%
Maximal term length: 15650
Weighting : term frequency (tf)

6.2 Most frequent words with 99% frequency accuracy:

Term document matrices are also very large for normal-sized datasets. Therefore, we provide a method to eliminate small expressions, i.e., expressions that occur only in a very small number of documents. In general, it dramatically reduces the matrix without losing the important relationships inherent in the matrix:

<<DocumentTermMatrix (documents: 109982, terms: 532)>>
Non-/sparse entries: 1993695/56516729
Sparsity: 97%
Maximal term length: 41

Weighting: term frequency (tf)

Transforming to lowercase

[1] "product version : <see about:version>urls (if applicable) :0.2.149.27other browsers tested: firefox / ieadd ok or fail after other browsers where you have tested this issue:safari 3: firefox 3: ok ie 7:okwhat steps will reproduce the problem?1. open any webpage on compaq 6715s running vista.2. try scrolling with the touchpad3. scrolling down will work , but up will not.what is the expected result?the page to scroll up.what happens instead?the page doesn't move.please provide any additional information below. attach a screenshot if possible.only a minor bug. "

Removing numbers

[1] "product version : <see about:version>urls (if applicable) :...other browsers tested: firefox / ieadd ok or fail after other browsers where you have tested this issue:safari : firefox : ok ie :okwhat steps will reproduce the problem?. open any webpage on compaq s running vista.. try scrolling with the touchpad. scrolling down will work , but up will not.what is the expected result?the page to scroll up.what happens instead?the page doesn't move.please provide any additional information below. attach a screenshot if possible.only a minor bug. "

Removing punctuation

[1] "product version see aboutversionurls if applicable other browsers tested firefox ieadd ok or fail after other browsers where you have tested this issuesafari firefox ok ie okwhat steps will reproduce the problem open any webpage on compaq s running vista try scrolling with the touchpad scrolling down will work but up will notwhat is the expected resultthe page to scroll upwhat happens insteadthe page doesnt moveplease provide any additional information below attach a screenshot if possibleonly a minor bug "

Removing stop words

[1] "product version see aboutversionurls applicable browsers tested firefox ieadd ok fail browsers tested issuesafari firefox ok ie okwhat steps will reproduce problem open webpage compaq s running vista try scrolling touchpad scrolling will work will notwhat expected resultthe page scroll upwhat happens insteadthe page doesnt moveplease provide additional information attach screenshot possibleonly minor bug "

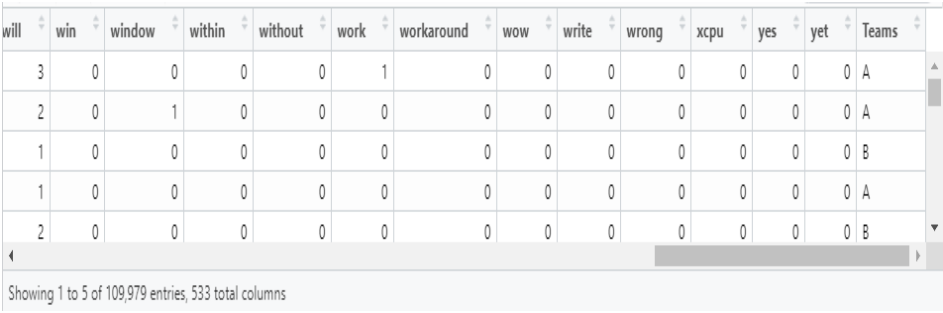
Stemming

[1] "product version see aboutversionurl applic browser test firefox ieadd ok fail browser test issuesafari firefox ok ie okwhat step will reproduc problem open webpag compaq s run vista tri scroll touchpad scroll will work will notwhat expect resultth page scroll upwhat happen insteadth page doesnt movepleas provid addit inform attach screenshot possibleon minor bug"

Remove White Spaces

[1] "product version see aboutversionurl applic browser test firefox ieadd ok fail browser test issuesafari firefox ok ie okwhat step will reproduc problem open webpag compaq s run vista tri scroll touchpad scroll will work will notwhat expect resultth page scroll upwhat happen insteadth page doesnt movepleas provid addit inform attach screenshot possibleon minor bug"

Figure 5. Transformation function in text document.



will	win	window	within	without	work	workaround	wow	write	wrong	xcpu	yes	yet	Teams
3	0	0	0	0	1	0	0	0	0	0	0	0	A
2	0	1	0	0	0	0	0	0	0	0	0	0	A
1	0	0	0	0	0	0	0	0	0	0	0	0	B
1	0	0	0	0	0	0	0	0	0	0	0	0	A
2	0	0	0	0	0	0	0	0	0	0	0	0	B

Showing 1 to 5 of 109,979 entries, 533 total columns

Figure 6. Converting DocumentTermMatrix as a matrix

7. Evaluation

Performance can be determined in various ways; However, accuracy, appeal, and precision are commonly used. **Table 1**, to determine them, we must first begin by understanding whether the document classification is True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN).

The confusion matrix is an important part of our study because reviews from datasets can be classified as duplicate or genuine reviews. The number of right and wrong estimates is summed with the values of the number and divided by the individual classes. This is crucial to the matrix of chaos. Calculating the confusion matrix will give us a better idea of what our taxonomic model corrects and what kind of errors it makes.

Table 1. Classification: TP, FP, TN and FN.

	Team-A	**Team-B**
Team-A	TP	FN
Team-B	FP	TN

Accuracy is commonly used as a benchmark for classification methods. However, accuracy values are much less likely to fluctuate in the number of correct decisions than accuracy and appeal:

$$Accuracy = \frac{(TP + TN)}{(TP + FP + TN + FN)}$$

Very often, the category of interest in text classification is exceedingly small. This high representation of the negative class in data retrieval problems causes problems in assessing the performance of classifiers using accuracy. In this case, the performance of the algorithm classification is measured by accuracy and subtraction.

Classification algorithms can achieve high accuracy by categorizing each example negatively—analysis of the scalability of several classifiers in classification experiments on text classification and noisy texts. Noise is the process of extraction (affected by errors) from media other than digital texts (e.g., transliterations of speech records collected by the recognition system). In a clean and noisy version of the same document (word bug rate between ~ 10 and ~ 50 percent), the performance of the

classification system is compared. Noisy lessons are obtained using handwriting recognition and optical character recognition simulation.

The mechanisms used to compile the classification set are i) the use of different subsets of training data in the same practice method, ii) the use of different training parameters in the same training method (e.g., the use of different starting weights for each neural network in the file) iii) the use of different learning methods. Many researchers have shown that combining multiple taxonomies improves classification accuracy by combining multiple taxonomies for text taxonomy.

8. Conclusion

The problem of text classification is a research topic of artificial intelligence, especially concerning the huge number of documents available in e-mails, discussion forum posts, and other electronic documents in the form of websites and other electronic texts. For the specified classification method, it was found that the classification performances of the classifiers differ based on the different institutions of the training texts; in some cases, such differences are quite significant. This observation indicates that the performance of the classification is to some extent related to its training corpus and that a good or high-quality training organization can obtain good performance classifications. Unfortunately, very little research has been found in the literature on using the training text corpora to improve classification performance. However, from the analysis using machine learning algorithms, we can predict which team (A or B) the newly reported bug can be assigned.

References

[1] Chaturvedi, K. K., & Singh, V. B. (2012). Determining Bug severity using machine learning techniques. 2012 CSI (CONSEG), 1–6. https://doi.org/10.1109/CONSEG.2012.6349519

[2] Zou, W., Hu, Y., Xuan, J., & Jiang, H. (2011). Towards Training Set Reduction for Bug Triage. 2011 IEEE 35th Annual Computer Software and Applications Conference, 576–581. https://doi.org/10.1109/COMPSAC.2011.80

[3] Anvik, J., Hiew, L., & Murphy, G. C. (2005). Coping with an open bug repository. Proceedings of the 2005 OOPSLA Workshop on Eclipse Technology EXchange, 35–39. https://doi.org/10.1145/1117696.1117704

[4] Čubranić, D. (2004). Automatic bug triage using text categorization. In SEKE 2004: Proceedings of the Sixteenth International Conference on Software Engineering & Knowledge Engineering, 92–97.

[5] T, K., Sekaran, K., D, R., V, V. K., & M, B. J. (2019). Personalized Content Extraction and Text Classification Using Effective Web Scraping Techniques. International Journal of Web Portals (IJWP), 11(2), 41–52. https://doi.org/10.4018/IJWP.2019070103

[6] Neethu, M. S., & Rajasree, R. (2013). Sentiment analysis in Twitter using machine learning techniques. 2013 (ICCCNT), 1–5. https://doi.org/10.1109/ICCCNT.2013.6726818

[7] Luyckx, K., & Daelemans, W. (2005). Shallow Text Analysis and Machine Learning for Authorship Attribution. LOT Occasional Series, 4, 149–160. http://localhost/handle/1874/296538

[8] Pushpalatha, M. N., & Mrunalini, M. (2019). Predicting the severity of closed source bug reports using ensemble methods. In Smart Intelligent Computing and Applications (pp. 589-597). Springer, Singapore

[9] Ramay, W. Y., Umer, Q., Yin, X. C., Zhu, C., & Illahi, I. (2019). Deep neural network-based severity prediction of bug reports. IEEE Access, 7, 46846-46857.

[10] Otoom, A. F., Al-jdaeh, S., & Hammad, M. (2019, August). Automated classification of software bug reports. In Proceedings of the 9th International Conference on Information Communication and Management (pp. 17-21).