# Applications of Data Mining Techniques in Software Engineering

[1]Lovedeep, [2]Varinder Kaur Atri

Department of Computer Science & Engineering
Guru Nanak Dev University Regional Campus, Jalandhar, India
Email: [1]Lovedeep1991@gmail.com, [2]varinder2002@yahoo.com

**Abstract: The field of software engineering concern with designing, developing, maintaining and modifying software. There are numerous types of data available in software engineering such as graphs, text, facts and figures. Meaningful information can be exacted from this complex data using well established data mining techniques such as association, classification, clustering etc. By uncovering hidden patterns using data mining software engineering data is made actionable. There are various goals in software engineering such as optimization, documentation, cost estimation etc. Selection of best mining method in each phase of software development life cycle helps in achieving these goals efficiently and the success rate of software is increased. Various software engineering tasks are improved using data mining techniques. In this paper, the focus is how data mining techniques helps in achieving the software engineering goals and benefit the software engineering tasks.**

**Keywords: Software Engineering, Data Mining, Software Engineering goals and tasks, Mining Software Engineering data.**

## I. INTRODUCTION

The software systems that we work with are inherently complex and difficult to conceptualize. This complexity lead to faults and defects as result increases the cost of software. Software metrics have long been a standard tool for assessing quality of software systems and the processes that produce them. But there are several drawbacks using the metrics as managers mostly rely on metrics which they can easily obtain and work with. Valuable metrics are difficult to obtain and are unavailable. The data generated in software system is huge and not easy to work with. If proper harnessing is done, it can be useful for various software engineering processes and phases. Due to large and complex data generated day by day at quite a high rate data mining is introduced in software engineering [1]. Software engineers are extensively applying data mining algorithms to various software engineering tasks so as to improve software productivity and quality. However mining software engineering data have several challenges and thus require number of algorithms to effectively mine text, graphs and sequences from such data. Software engineering data includes execution traces, historical code changes, code bases, mailing lists and bug data bases. Software engineering data contains a wealth of information about a project’s status, progress, and evolution. Using well-established data mining techniques, engineers and researchers have started exploring the potential of this valuable data to better manage their projects and to produce higher quality software systems that are delivered within budget and specified time period. Data mining is used by software engineers to previously unknown and unique data statistics within a set of collected data. Data mining tools are useful in predicting the future trends and behaviors which are helpful for engineers to take proactive knowledge driven decisions.

This paper is organized as follows. In section II goals of software engineering have been described of the paper. In section III the various techniques has been discussed. Tasks of software engineering have been discussed in section IV for direction to emerging researchers and section V and VI gives a summary and conclusion of the paper.

## II. GOALS OF SOFTWARE ENGINEERING

There are some goals in software engineering. Data mining is quite useful in achieving these goals efficiently and quickly as well.

A. To find and fix bugs: Software bug estimation is a very essential activity for effective and proper software project planning. All data related with software bug is kept in software bug repositories. A software bug repository contains interesting information related to the development of a project. Data mining techniques can be applied on these repositories to uncover useful and interesting patterns. A prediction data mining technique is used to predict the software bug estimation from a software bug repository. Initially the summary and description of bug for which estimation is required, is matched against the summary and description of bugs that are available in bug repositories. To match the summary and description for any pair of software bugs we use weighted similarity model. Next step is to calculate the fix duration of all the similar bugs and its average is calculated, which gives the predicted estimation of a bug [2].

B. Documentation: software document data is high important but it is complex in nature processed by data mining techniques. Source code, system administration and application documentation consists large buffer of documents and free text for mining and software analysis. External and internal documentation also play the important role for data sources. The types of documents(html, portable document format, text etc) available in large variety and another important sources are the multimedia data(audio, video figures) [3].

C. Software configuration management data: software configuration management a system (SCMs) includes documents, software code, status accounting, design models, defect tracking and also include revision data. In SCMs the large amount of data is available and most valuable data is kept from different sources and the most of the SCMs data is in structured form [4].

D. Source code: source code is a important source for data mining in software engineering. The various applications of data mining in software engineering is program comprehension, maintenance and software components analysis. Initially available source code is always a caveat and parses source code language available and it can be seen as structured form. Applying data mining techniques in source code includes predicting change propagation, change history, predicting defect densities in source code files.

E. Mailing lists: large software systems especially open source software bridging developers and users. Mailing lists contain hard data contain a lot of free message, text and author graphs. Data mining applications are not limited to text analysis, linguistic analysis and text clustering of subjects.

F. Cost estimation: cost estimation is a approximate judgments of the cost for a project and it is the one of the main problem in software engineering [5].there are too many variables involved to calculation for the cost estimate( technical, human, political and environmental) and measure in terms of efforts and metric used is person months or year. Accurate cost estimation is very important for every kind of project and estimated by cocomo model.

## III. TECHNIQUES

A. Association rule: Association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. Association Rule mining techniques is applied to the records in order to discover the patterns that are likely to cause high severity defects [6]. The discovered patterns are then helpful to predict the subsequent actions that may result in high severity defects. The concept of association in mining software engineering data is based on set of strong rules.The computational cost of association rules mining in software development can be reduced by reducing the number of passes over the database, sampling the software database, adding extra constraints on the structure of patterns, through parallelization [7].Association mining searches for strong and interesting relationships among a large set of data items. After discovery of interesting association relationships among huge amounts of software and cost records, it can be helpful in a many decision making process, such as catalog design, cross- marketing, and loose-leader analysis [8]. An example of association rule mining is market based analysis and requirement analysis for developing new software. This process analyzes customer requirements and interests by finding association from past interactions with customers and products of competitors. Association rule based algorithms are developing swiftly. Algorithms such as apriority algorithm, FP-growth algorithm and OPUS search are quite helpful in generating association rules that are utilized in software engineering.

B. Classification: Classification is one of the main tasks in data mining for assigning a data item to a predefined set of classes. Classification can be described as a function that maps (classifies) a data item into one of the several predefined classes [9]. Here the goal is to induce a model that can be used to classify future data items with unknown classification into unique classes. In software development process the performance of classifier depends upon the type and class of data. There are different forms of data available in software engineering. It is imported to work with relevant data items and classify them into sub classes and keep on adding new data items into pre existing classes. We implement different classification algorithms in software engineering to solve various problems in different phases. Classification can be used to identify the types of bugs and thus helps in building bug detector. Decision trees are important tools in classification technique that helps in identifying the risky modules in software depending upon the attributes of system and modules. Classification and assignment can sometimes be automated, but are often done by humans, especially when a bug is incorrectly filed by the reporter or the bug database.

C. Clustering: Cluster analysis is a group of multivariate techniques whose primary purpose is to group entities based on their attributes. Similar objects are placed in the same cluster according to predetermined selection criteria. The objective of any clustering algorithm is to sort entities into groups, so that the variation between clusters is maximized relative to variation within clusters [10]. The set of entities to cluster needs to be identified, before applying clustering to a software system. The next phase is attribute selection. Most software clustering methods initially transform a fact base to a data table, where each row describes one entity to be clustered. Each column contains the value for a specific attribute. After completion of all preparation steps the clustering algorithm can start to execute. Clustering algorithms used in software engineering are: graph-theoretical algorithms, construction algorithms, optimization algorithms, hierarchical algorithms. For high-dimensional data, many of the existing methods fail due to the curse of dimensionality, which renders particular distance functions problematic in high-dimensional spaces which led to new era of clustering algorithms for high-dimensional data that focus on subspace clustering and correlation clustering that also looks for arbitrary rotated subspace clusters that can be modelled by giving a correlation of their attributes. Examples for such clustering algorithms are CLIQUE. Several different clustering systems based on mutual information have been proposed. One is Marina Meilă's variation of information metric [11] another provides hierarchical clustering [12]

D. Text mining: Approximately 80% of information is stored in computers is in form text [13]. Example of software engineering text data includes project and bug reports, e-mails and code comments Text mining is an area of data mining with extremely broad capability. Rather than requiring data in a very specific format such as numerical data, database entries, text mining can discover previously unknown information from textual data. As many artefacts in software engineering are based on text, there are abounded sources of data from which information may be extracted. There are several applications of text mining and their implications for software development processes. Code duplication is one of the biggest problems which complicates maintenance and evolution of software systems. Several drawbacks of all existing code duplication techniques can be overcome by using visual approach which is language-independent. Although this approach requires no language-specific parsing, it is able to detect significant amounts of code duplication. Also duplication of bug reports is common in software development, it can be overcome using neural language processing with data mining [14].Text data mining refers to the discovery of hidden information and potentially useful knowledge from a collection of texts which is done by automatic extraction and by analyzing information [15]. A key factor is to extract the appropriate information and link it together to form new facts to be explored further. The Natural Language Description Technique combines computer science and linguistics to enhance the interactions between computers and natural languages.

E. Metaheuristic: In software engineering the engineers focus not only finding the solution of problem but an acceptable or near optimal solution from large number of alternatives. The task of selecting the best solution is not easy as change in one single parameter have large impact on final product. Metaheuristic techniques are solution to these problems regarding the software engineering. The techniques provide set of generic algorithms that are helpful in searching for optimal or near optimal solution to a problem within a huge multi-modal search space. Many problems regarding software engineering can be easily reformulated as search problems and metaheuristic search algorithms can be applied. The six area of software development ranging from scheduling and requirements, cost and effort estimation, system integration to transformation of source for maintenance and re-engineering are improved using three local search techniques i.e. tabu search, simulated search and hell climbing [16].

## IV. TASKS IN SOFTWARE ENGINNERING

Role of data mining in improving effectiveness of software engineering tasks

A. Development tasks: Software development is a creative process as no two programs are the same. It is difficult to accumulate enough relevant data in the initial programming phase of a software project which provide insights that is helpful in guide development. With use of dynamic analysis and mining of revision histories, bugs can be fixed with constant check-ins. The errors found using this approach in development phase are mostly previously unknown [17].

B. Management tasks: Managers can utilize the historical data and software artifacts to improve the management tasks. It becomes even important for manager dealing with extremely large projects as problems such as bug prediction and resource allocation arise and data mining provides many innovative solutions. Use of software tools improves the quality of software but for large projects and organizations it becomes expensive and hard to manage and maintain. Data mining helps in cost based analysis and selection of proper tool depending upon the tool usage statistics with estimates of developer effort [18].

C. Research tasks: The goal of data mining from point of view of engineering researcher is to gain understanding about a number of projects which is helpful in characterizing patterns in software development. Researchers generally analyze data from open-source projects, but mining data from organizations like Sourceforge.net is fraught with drawbacks such as dirty data and defunct projects. Software evolution is favorite and latest topic for software data miners. A better way to understand a program's development history is by making use of partitioning and clustering of version data [19].

## V. SUMMARY

| S. NO | DATA MINING METHODS AND TECHNIQUES | FIELD OF APPLICATION IN SOFTWARE ENGINEERING | TYPE OF DATA SET |
|---|---|---|---|
| 1. | Association rule | Catalog design, prediction of severe defects, cross | Large variable data set, numerical, alphanumeric, visual and audio data |

| | | | |
|---|---|---|---|
| | | marketing | |
| 2. | Clustering Technique | Development of cost effective tools, discovery and localization of program failures | Statistical data, discrete and comparative numerical data. |
| 3. | Classification | Bug tracking, discovery and maintenance of risky modules | Graphical data, trees and graphs |
| 4. | Text mining | Detection of code duplication, bug duplication reports, | Project reports, bug reports, codes, emails, text data |
| 5. | Metaheuristic | Cost and effort estimation, maintenance , scheduling and requirement analysis. | Statistical data, large and numerical values visual data sets |

## V1. CONCLUSION

As software engineering generates huge amount of data, it is important to utilize it properly so that the problems regarding the software development cycle can be solved efficiently. Some extreme problems are faced in software engineering field, such as occurrence of bugs, rise in cost of software maintenance; unclear requirements, etc. that can effect software productivity and quality. The paper outlined some of the data mining techniques which can be applied to different types of software engineering data in order to solve the challenges posed by software engineering tasks such as development, management, debugging, and maintenance. Also data mining is as good as results it produces so quality and quantity of available data and computational cost determines the success of data mining in software development process. The engineers and data miners should carefully employ mining techniques so as to cut down cost of tools and gain more support from data available. In coming time the research is likely to be carried out in field of increased automation and achieving even higher simplicity.

## REFRENCES

[1] Q. Taylor and C. Giraud-Carrier, “Applications of data mining in software engineering”, Int. J. Data Analysis Techniques and Strategies,2010.

[2] N. Nagwani and S. Verma, “Predictive data mining model for software bug estimation using average weighted similarity”, In proceeding of: Advance Computing Conference (IACC), 2010.

[3] A. E. Hassan. “The road ahead for mining software repositories”, in Proceedings of the Future of Software Maintenance at the 24th IEEE International Conference on Software Maintenance,2008.

[4] Z. Li and M. Reformat, “A practical method for the software fault prediction”, in proceedings of IEEE International Conference Information Reuse and Integration (IRI),2007.

[5] C. Elkan. The foundations of cost-sensitive learning. In Proceedings of the Seventeenth International Conference on Machine Learning,2001.

[6] C. CHANG and C. CHU, “Software Defect Prediction Using Intertransaction Association Rule Mining”, Int. J. Soft. Eng. Knowl. Eng,2009.

[7] S. Kotsiantis and D. Kanellopoulos, “Association Rules Mining: A Recent Overview”, GESTS International Transactions on Computer Science and Engineering, 2006.

[8] N. Pannurat, N. Kerdprasop and K. Kerdprasop “Database Reverse Engineering based on Association Rule Mining” ,IJCSI International Journal of Computer Science Issues,2010.

[9] U. M. Fayyad, G. PiateskyShapiro, P. Smuth and R. Uthurusamy, “Advances in Knowledge Discovery and Data Mining”, AAAI Press, 1996.

[10] M. Shtern and Vassilios, “Review Article Advances in Software Engineering Clustering Methodologies for Software Engineering”, Tzerpos Volume,2012.

[11] M. Marina, "Comparing Clusterings by the Variation of Information". Learning Theory and Kernel Machines. Lecture Notes in Computer Science,2003.

[12] K. Alexander, S. Harald, A. Ralph; Grassberger, Peter, “. Hierarchical Clustering Based on Mutual Information”, 2003.

[13] M. Gegick, P. Rotella and T. Xie, “Identifying security bug reports via text mining: an industrial case study,” 7th IEEE Working Conf. Mining Software Repositories (MSR), 2010

[14] P. Runeson, and O. Nyholm, “Detection of duplicate defect reports using natural language processing”, in Proceedings of the 29th International Conference on Software Engineering,2007..

[15] G. Vishal and S. L Gurpreet, “A survey of text mining techniques and applications,” Journal of

Emerging Technologies in Web Intelligence,2009.

[16] C. Kirsopp, M. Shepperd and J. Hart "Search Heuristics, Case-Based Reasoning and Software Project Effort Prediction", In proceeding of: GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference,2002.

[17] B. Livshits and T. Zimmermann, "Dynamine: finding common error patterns by mining software revision histories", ACM SIGSOFT Software Engineering Notes,2005.

[18] D. Atkins, and A. Mockus, 'Using version control data to evaluate the impact of software tools', in Proceedings of the 21st International Conference on Software Engineering,1999.

[19] J. Howison and K. Crowston, "The perils and pitfalls of mining sourceforge", in Proceedings of the International Workshop on Mining Software Repositories,2004.

◈◈◈