# LOG FILES UTILITY FOR SOFTWARE MAINTENANCE

**Wasim Khan, Hannan Ansari, Anwar Ahamed Shaikh**

***Abstract*— Software Engineering data is very huge and so it attracts many researchers for research on this data. Advanced Software Systems are heavily relied by the companies associated with the software development. There is a strong race among the software companies to develop the better software resulting in an urgent requirement to improve the performance and quality of the software. In this paper, to improve the maintenance of the software, we focus on the application of Data Mining algorithm on software engineering data such as execution trace log files. Most accessed data is identified from using the Data Mining Alogrithm so that we can identify the sensitive part of the code and after identifying this error prone code, errors and defects can be minimized by emphasizing more on this code resulting in the software system performance improvement.**

***Index Terms*— Software Engineering, Software Quality Attributes, Software Maintainability, Execution trace Log File (Log File), Data Mining, Data Mining Algorithm, Frequent Pattern Mining Algorithm, Log Parser, Logger Level.**

## I. INTRODUCTION

Now a Days Software Engineering is an approach for software system development so that our efforts can be applied in an efficient manner. This efficient manner is an important step to achieve the expected results in a highly optimized way. Mainly it is concerned with the different processes of the development of software system, various methods and tools to support the production of software. It is also concerned with achieving the software quality within the budget and schedule which have been already defined in the project plan[7] but it is very difficult in terms of practical implementation. As we know there is a strong competition among the Software development companies to include much functionality in a short period of time and it results in the attempt to comprise with the quality of the software. So there is an urgent need for software quality Maintenance.

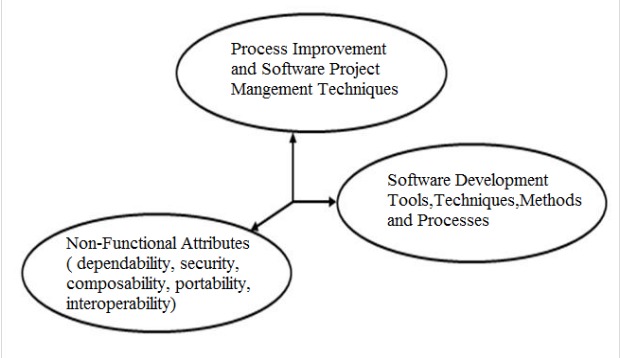
Reliability,Maintenance,Testability,Usability,Portability,Correctness,Efficiency etc are the attributes which represent the predicted behavior of a system within the conditions for which it was supposed to be developed. But we are addressing here only one but an important software quality attribute, Maintenance, an important concern for the software development companies. Maintainability strongly supports the improvement of the quality and performance of the software Maintenance makes it easy to correct the defects or making a change in the software so it is assumed to be an effort to maintain the software which in turn enhances the software development processes.
Software Engineering data is very huge so it attracts many researchers to research on this data. The problem of improving

the quality and productivity can be solved by working on the Software engineering data because there is a flood of the software engineering data and it is very useful to solve various problems.

So keeping in mind the aim of providing a straightforward and economical approach so that the required software quality is achievable, we tend to propose a brand new technique through which we can get surely the satisfactory and convincing solution. The proposed technique is actually introductory but very clear. In this Research, sequence data is targeted for analysis. We would impose the Data Mining algorithm on execution trace log files which are generated by the activities performed by the end user in real time. In this analysis, Execution trace log files are treated as software engineering data.

## II. BACKGROUND

In this current age of Automation, Software Engineering is a strong field for research and analysis. Software Engineering can be considered in three dimensions where one selective aspect is concerned by a particular dimension. All of the tools, techniques, methods, and processes which are actually required to develop the software are covered by First Dimension. Various management techniques which are required for organizing the software projects are covered by the second Dimension. Apart from covering the management techniques, additionally it monitors how much the development is effective and boosts the development process. The third Dimension takes care of the way in which the non-functional attributes of the software will be achieved. Non-functional attributes have nothing to do with the function of the software however they focus on the way on which it does it.



There are various data mining techniques displayed by an intensive survey and these techniques stresses to apply on software engineering data[3]. Researchers can inspect the potential of this useful software engineering data so that the software projects can be easily managed and quality of the system can be improved to the higher extent resulting in the well budgeted and timely projects.

The performance of a software product is evaluated by the Software Quality Attributes such as Availability,Reliability,Serviceability,Usability,Efficiency,Portability,Scalability,Security,Recoverability,DurabilityMaintainability,Dependability,Supportability,Robustness,Performance etc. Among these attributes some extra efforts are made to attain some attributes which are more emphasized in relation to software requirements such as Reliability, Availability, Supportability, Performance, Maintainability, Usability etc.

Numerous algorithms are suggested which can be applied on the software engineering data. The Common Algorithms for sequence databases are the Apriority Algorithm, GSP(Generalized Sequential Patterns),Free Span, Sequential Pattern Discovery using Equivalence classes(SPADE), PrefixSpan and the more-recent FP Growth technique. With the help of approaches like BFS/Apriority Approach, DFS/Pattern Growth Approach, Diagonal Approach etc, graphs can be mined[2].Text mining algorithms are text generalization, clustering, classification approach etc,. The particular kind of algorithm can be used depending on the software engineering data [1].

Out of the many phases of software life cycle, Software Maintenance is taken into account as a very complicated and necessary stage which consumes almost half of the total allocated effort to a software project[9] [10].In the past, Numerous Experiments have been made to apply the data mining algorithms so that the maintenance can be enhanced resulting in overall improving the performance of the software project.

## III. MOTIVATION

Starting from the software requirements and then going through the design, coding, testing, and maintenance phases, all are human pivoted activities[5]. There are some types of code which are more sensitive in comparison to the rest of the code and which may make the software system defected in the future. So there is a strong need to pay some additional attention towards this code which is more sensitive rather than going through the entire code. Logger level can be used to identify this senstive type of code.

The Information about a program's execution are recorded using the Execution trace log files. When any type of defect occurs in the system these log files are accessed normally to diagnose problems with the software but are not analyzed regularly. So there is a need to motivate the use of these execution trace log files regularly for analysis for the software quality improvement.

## IV. CURRENT SCENARIO

The defects are dealt by the programmer responsible for maintenance of the software. First of all every request is thoroughly checked by the programmer. This is then followed by the investigation of the validity and if it is confirmed the next step is suggesting a solution for the request after analyzing it. Finally programmer applies the changes by getting the permission from the authorities.

Every Maintenance Programmer uses unique processes, practices and activities such as Transition, Modification and Contracts[8].

1)A system is handed over step by step to the Maintenance programmer by the integrated and controlled sequence of activities.[TRANSITION]

2)Maintenance Programmer maintains the particular contracts and Service level Agreements after proper negotiation.[CONTRACTS]

3)Maintenance programmer employs a problem solving method for the prioritization of documents for the setting the route for requests.[MODIFICATION]

## V. CURRENT APPROACH

There are some steps involved in Software Engineering data mining. First step involves the Collection of the Software engineering data and its analysis. Then the next step is to choose a task of software engineering. These Two steps are parallel. The next step is to preprocess the data with the help of techniques like Extraction, Cleaning, Integration, Transformation and Data Reduction etc to improve the quality of the software engineering data and the result of mining algorithm.

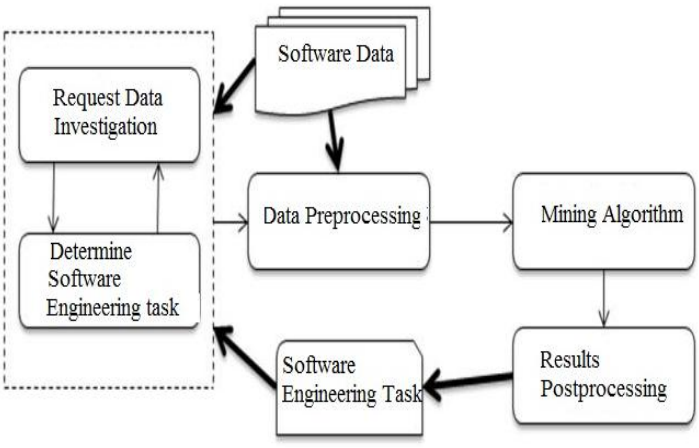



Fig2. Data Mining Architecture for software Engineering Data

The next step is choosing a mining algorithm which fulfills the investigated requirements. When the Dataset are generated from the preprocessed data stored in the database table, software engineering data can be mined using any suitable data mining algorithm depending on the software engineering task. At Last, the output is remodeled into correct format which in turn would help the software engineering task.

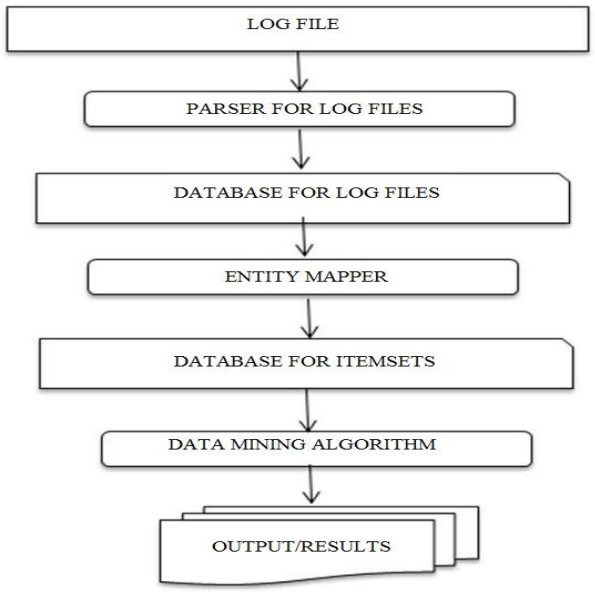
  

## VI. PROPOSED APPROACH

As we know that software system execution creates logs and initially these logs are saved with in a file. Log Parser is used to preprocess the log files so that Data mining algorithm can be applied on these files and storing these files into a database table. So the logger level, package, class, method and log message is stored by the Log Parser. Every entry of the log file is represented in the form of database transactions in the database table. There is no need to repeatedly use the log file parser because now we are handy with the data for the application of data mining algorithm on it. Packages, Classes and methods are mapped to a distinct identifier because they are now identified from the database tables.

It is a tough job to scan the Package, Classes and methods long names. Therefore we have used the Mapping to make the task of applying the data mining algorithm easy.

Data fields are combined to form Item Sets after the completion of mapping to make them ready for the application of data mining algorithm.




Now the item sets are mined with the help of Frequent Pattern Mining Algorithm and these Item sets which are occurring frequently can be identified. Then to help the software engineering task we tend to post process the results into the software engineering data. Error prone codes are identified by using the logger level and then we are having the names of the packages, classes and methods repeatedly occurring. So these lines of code are emphasized more resulting in overall improvement of maintenance of the software because in future we are able to make the changes in the software.

## VII. ASSUMPTIONS

Following assumptions have been made during this proposed approach:

1.There is a proper logging of errors.

2.Format of each line in log file is same.

3.Proper Date wise Maintenance of logs

4.Acceptable logging for all Classes.

5.Specified name format is given to execution trace log file.

6.There are fixed number of items in each transaction.

## VIII. IMPLEMENTATION

To implement the proposed approach, Execution trace log files are used of a module responsible for receiving account information of a web application. Financial Accounts of the clients are managed by this application.

Crucial actions associated with the clients such as payment, debit, credit etc are handled by this application therefore maintenance of such crucial application needs big efforts.

We have taken log files where multiple users are using the system in production.

*A. Log Parser*

To design the data mining algorithms, vast amount of knowledge is required. The software engineering professionals could have lack of this information whereas the application of data mining in software engineering may be overlooked by the Data mining experts. So there is a strong need of a data format which use is quite easy so that Data mining experts can analyze software engineering data.

The above stated problem can be solved by using the Log Parser. Execution trace Log Files are parsed by scanning the particular line of the log files and then fields and records of the table are tokenized. So the logger level, packages, classes, methods and log messages with relevance a selection execution trace. Then we preprocess this data so mining algorithms can be applied on it more effectively.

Entries in the log files are shown below:

[DEBUG] [Sep 20 05:45:26] [Thread-26] [mentor.revmgmt.AR1GeneralLogHandler.mentor.ar.datalayer] [Abstract PM.<init>] Setting operator Id - batch: 0

[DEBUG] [Mar 20 05:45:26] [Thread-25] [mentor.revmgmt.AR1GeneralLogHandler] [PartitionHelperBaseCustomization.setTransactionLogPartitions] PARTITION_PAR == 1

mysql> select * from logdata limit

id: 1

date: 2015-09-20 05:45:26 logger_level: DEBUG

package_name:mentor.revmgmt.GeneralLogHandler.mentor.ar.datalayer

class_name: AbstractPM method_name: <initialization>

log_message: SettingopId-batch:1

*B. Data Mining Algorithm*

Patterns that mirrors completely different levels of data are generated by Various mining algorithms and depending on

the needs of the particular mining, specific data mining algorithm is chosen. The code which is more accessed can be identified here and pattern mining algorithm can be chosen. The repeatedly occurring packages, classes and methods are identified with the help of the chosen frequent pattern mining algorithm.

Apriori Alogrithm is used for mining frequent data and it is based on the rule that any subset created by the frequent data-set must also even be a frequent data-set. Method invocation relies on the class invocation which in turn relies on the package which act as a container of the classes. So a method which identified as repeatedly occurring the involved packages and classes would even be accessed often times. The package and class correlates with a specified workflow of the software so we are getting the workflows as a result of the data mining algorithm and these workflows are occurring repeatedly in the system execution. We identify the inaccurate workflows after investigating the logger levels result an then paying the extra attention to these workflows.

We have thought of the trace log file containing over 10K transactions. After scanning of the file, it is parsed into the transactions and then they are mapped into transactions. We assume that Pn, Cn, Mn, Ln are respective mapped names of Packages, Classes, Methods and Logger levels name. We are considering only a set containing 10 item-sets to demonstrate the working of proposed approach. Following table is representing the item sets.

| ID | Item-Sets |
|---|---|
| Id1 | P1, C1, M1, L1 |
| Id2 | P2 ,C2, M2, L1 |
| Id3 | P1, C1, M1, L1 |
| Id4 | P1, C3, M3, L1 |
| Id5 | P2, C4, M4, L1 |
| Id6 | P2, C5, M5, L1 |
| Id7 | P2, C5, M5, L1 |
| Id8 | P3, C6, M6, L1 |
| Id9 | P1, C1, M1, L1 |

[p1], [p2], [c1], [c5], [m1], [m5], [l1] are the frequent itemsets[size=1],generated by the application of Apriori algorithm on the itemsets.Itemsets[size=2] are generated with the help of itemsets [size=1].On Snipping these itemsets we get the next frequent itemsets, [c1, l1], [c1, m1], [p1, c1], [c5, m5], [p2, c5], [m1, l1], [m5, l1], [p1, l1], [p2 , l1], [p1, m1], and, [p2, m5]. Then the next set is generated by using the items of the above itemsets.[c1, m1, l1], [p1, c1, l1], [p1, c1, m1], [p2, c5, m5], [p1, m1, l1], and [p2, m5, l1] are the obtained itemsets[size=3].At Last,itemsets[size=4] are formed by combining this and [p1, c1, m1, l1] is the final frequent itemset.

We would target the last set of the frequent itemsets because we want the frequent workflows in the execution.So the Packages,Classes,Methods and logger levels are contained in this set.The itemset are mapped once more to represent it in terms of the software engineeering data and for that Corresponding Packages,Classes,Methods and logger level original names are used to replace the unique identifiers previously used.Now final results are generated that will help the maintenance programmer by identifying and focusing more on the emphasized code.

[p1, c1, m1, l1] is the final frequent item set and now replaced by the original name. P1 is the name of the package used for <<mentor.revmgmt.AR1GeneralLogHandler.mentor.ar.datalayer>>. <<AbstractPM>> is represented by C1. M1 is name of the used for <<init>>. logger level <<Debug >> is represented by L1.So after replacing we get the final result as follows:

The frequent execution workflow considering 10 transactions from execution trace log file is as below:

Package Name : mentor.revmgmt.AR1GeneralLogHandler.mentor.ar.datalayer
Class Name:AbstractPM

Method Name: init

Logger_level:Debug

## IX. CONCLUSION

We have focused on improving the Software maintenance process. The proposed approach tries to extend the utility of trace log files. End users generate the trace log files while performing the activities in real time. The technique of application of data mining algorithm on this data to improve the quality of the software is very useful since these data are generated by the end users performing the transactions. so this would lead to the error free transactions between the software and end user.

When an error is detected, manual efforts are made to analyze the trace log files but the usage mining algorithm reduces it. The maintenance programmers will work to improve only the code which is identified as the sensitive more error-prone code. The modules containing this sensitive code are checked regularly not only at the time when an error is occurred. The Quality of the software is improved by using the maintenance approach based on prediction.

## X. REFERENCES

[1] Tao Xie, Suresh Thummalapenta, David lo, Chao Liu,"Data Mining for Software Engineering", IEEE Computer, August 2009, pp. 55-62.

[2] A.V.Krishna Prasad, Dr.S.Rama Krishna, "Data Mining for Secure Software Engineering - Source Code Management Tool Case Study", International Journal of Engineering Science and Technology, Vol. 2 (7), 2010, 2667-2677.

[3] NATO Science Committee, "Software Engineering", Report on a conference, Garmisch, Germany

[4] Manoel Mendonca, "Mining Software Engineering Data: A Survey, DACS State-of-the-Art Report", University of Maryland, Department of Computer Science, Nov 1999.

[5] Prof. D. Vernon, "Course Notes", Khalifa University http://www.vernon.eu/courses/David_Vernon_Software_Engineering_Notes.pdf

[6] Software Maintenance and Re-engineering, CSE2305 Object-Oriented Software Engineering, http://www.csse.monash.edu.au/~jonmc/CSE2305/Topics/13.2.5.SWEng4/html/text.html

[7] Tao Xie, Jian Pei, Ahmed E. Hassan, "Mining Software Engineering Data".

[8] Alain April, Jane Huffman Hayes, Alain Abran, Reiner Dumke, "Software Maintenance Maturity Model (SMmm): The software maintenance process model", J. Softw. Maint. And Evolution 2004.

[9] Pigoski T.M., Practical Software Maintenance: Best Practices for Managing your Software Investment, Wiley Computer Publishing, 1996.

[10] Sommerville, Software Engineering, 6th ed., Harlow,Addison-Wesley, 2001.

[11] http://www.cs.helsinki.fi/u/langohr/graphmining/slides/chp1.pdf.

We have aimed to enhance the maintenance process for a software system. The research work intends to increase the utilization of the execution trace log files. The execution trace log file is generated by the activities performed by the end user in real time. Applying mining algorithm on this data to enhance the quality of the software application proves beneficial since these are the transaction performed by the end user. This would result in data error free interactions of the end user with the software application.

**Wasim Khan** completed B.Tech(IT) in 2006 and M.Tech in 2011 has vast experience in the area of "Big Data", Data Mining, IDSs, Software Engineering, Network Security , WSNs etc. He has experience of more than 9 years in teaching and also published many research paper in various fields, currently working as a Assistant Professor in department of Computer Application of Integral University Lucknow U.P.

**Hannan Ansari** completed B.Tech(IT) in 2012 and M.Tech KIIT university in 2015 has experience in the area of IDSs, Software Engineering, Network Security, WSNs etc. He has total experience of 1 year and 6 months in Software Industries in Web Development and Designing and more than1 year experience in teaching and also published three Research papers in various related fields currently working in Integral University Lucknow U.P.

**Anwar Ahamed Shaikh** is B.Tech and M.Tech has vast experience in the area of "Big Data", Data Mining, IDSs, Software Engineering, Cloud Computing etc. He has experience of more than 7 years in teaching and also published many research paper in various fields, currently working as a Assistant Professor in department of Computer Science and Engineering of Integral University Lucknow U.P.