International Journal of Information Technology and Management

Vol. IX, Issue No. XIII, August-2015, ISSN 2249-4510

THE IMPLEMENTATION OF DATA MINING APPROACH FOR TESTING

AN INTERNATIONALLY INDEXED PEER REVIEWED & REFEREED JOURNAL

www.ignited.in

International Journal of Information Technology and Management
Vol. IX, Issue No. XIII, August-2015, ISSN 2249-4510

The Implementation of Data Mining Approach for Testing

Sheo Kumar[1] Dr. Rajan Anand Malik[2]

[1]JJTU Scholar, Jhunjhunu

[2]Director JEMTEC Greater Noida

Abstract – In this paper we present about the implementation of data mining approach for testing. The design of an appropriate test suite for software testing is a challenging task. It requires a suitable tradeoff between effectiveness, e.g., a sufficient amount of test cases to satisfy the test goals of a given coverage criterion, and efficiency, e.g., a redundancy-reduced selection of test cases. Recent test suite optimization approaches, therefore, usually require an explicit enumeration of existing test cases.

Keywords: Software Testing, Test Cases

----------X----------

INTRODUCTION

Software engineering [1] is a field of study related to designing, implementing and modifying software to build it affordable and maintainable. In this process Software testing [1] is one of the main processes of software engineering. Software testing [1] is one of the indispensable areas of the software development life cycle. It is generally conducted on large scale to find the errors in our software and also to test the software for correct results. It aims at providing assurance to the client that our software works fine at any circumstances. This requires building test cases that exploits each and every possible route of the software. It is very tedious to explore each and every possible test case manually. Automated test case generation [2] has already been started where test cases are built automatically. This generates thousands of test cases through a simple program at a faster rate. But the problem with the above approach is if the software is built on thousands of lines of code, for execution of each test case it takes a lot of time to know the output and if the test suite is more the execution of the test suite [3-5] may take even days to complete.

REVIEW OF LITERATURE:

Test suite contains test cases that are machine generated. It contains redundant test cases too. Our approach particularly deals with the above issue. Testing these redundant test cases even increases the time taken by testing phase. With increase in number of test cases this amount of time is significant. We have used data mining [6] approach to deal with the above issue. Data mining [6] is a novice research area where it is intended to extract patterns out of data that are not visible. In Data mining we have use clustering technique [6-7] which clusters similar data. The application of data mining techniques to our test suite significantly reduces the test suite. The coverage [8] either path or conditional by the reduced test suite yielded good results.

SOFTWARE TESTING:

Testing is indispensible phase of software development life cycle. Debugging [3] is the search for cause of defects. Testing leads to uncovering problems which enhances further debugging. Software deployed without testing leads to unreliability. Hence testing the software to the full extend is a necessary task while building a software. Testing the software implies executing possible test cases. Extend of testing can be evaluated using several techniques like path coverage, conditional coverage, code coverage etc. This phase of Software Development Life Cycle [3] is the most expensive phase. It requires lot of time and effort. Hence optimization of test cases is a must. But first we need to see how a test case looks like. A test case [2, 4] is a collection of different attributes of the software. Attributes are the inputs to the software. So a test case can be compared to a touple in a database table. It has ID, attribute1, attribute 2…attribute n. A good test case is one that is able to find faults with the software. Hence the output of test case is a pass/fail. A test suite [2, 4] is a collection of automated generated test cases for particular software. But due to the process of automation redundancy can be initiated in the process of test data generation. Redundancy is the repetition of data, between one test case and the other. So optimization

of test suite is important to achieve by which lot of time can be saved from executing redundant or unnecessary test cases. The behavioral patterns exhibited by the test suite helps us in this process of automation Due to the development in software processes; a lot of automation work is carried out in all of its activities. Like so automation is carried out in generating test cases also which enhanced the functionalities in testing phase. An automated test data generator is a program built with the feature of generating inputs to the software by considering business rules and input domain [10].In spite of a lot of care taken in generating test cases significant amount of data is replicated in different test cases. This replicated data isn't visible enough to capture unless and until we use sophisticated techniques like data mining.

- **Data mining:**

Data mining [11-12] is a semi-automated process of finding patterns in the data. It is basically knowledge discovery in data. This knowledge discovered can be represented by a set of rules, equations relating different variables and other mechanisms of predicting outcomes.

The manual component of data mining [13] is the preprocessing phase where data is prepared acceptable by the algorithms and post processing phase involving discovering patterns to find out new ones that are useful. There are three main techniques in data mining classification, association rules and clustering [13]. Classification is a technique that classifies data into different classes by building models like decision trees. By using these models it predicts the behavior of future data. Association rules are the techniques used to find relationships or associations between different entities of an instance. With these associations we can predict the nature of one when the other changes. Clustering [14] is a technique used in finding clusters of points in the given data. In other words clustering is grouping together similar points into a single cluster. This behavior of grouping can be found out by different metrics like distance, density and grid based approaches. Within a cluster all set of points in that cluster are found to have similar behavior. In order to ease this process of data clustering, in the next section we introduce a tool called weka[15] which helps us in filling the gap between the process of software testing and knowledge mining.

- **Weka:**

Weka[13] is an online freeware tool for data mining. It has implementations for different approaches to data mining. There are various clustering algorithms available like K-means, DBSCAN [10], etc. But for this study we considered K-Means [17] which is a better way to restrict the number of clusters required depending on the value of K.

- **k-Means algorithm:**

One of the most widely used clustering algorithms is K-Means clustering [14]. This minimizes the mean squared Euclidean distance from each data point to its nearest centre.

Here we have a good control upon the number of clusters produced. So while reducing the test suit depending upon the number of test cases we wanted we can fix the value of k. Given a database

$$D = \{t1,t2,\ldots,tn\} \quad (1)$$

Tuples and an integer value k , the k-Means algorithm defines a mapping

$$F : D\text{->}\{1,2\ldots k\} \quad (2)$$

[1]. A cluster ki contains all the tuples that are mapped to it.

Step1: Randomly pick k points as centroids of k clusters.

Step2:

- For each point assign the point to the nearest cluster.
- Re compute the cluster centroids.
- Repeat Step2 (until there is no change in clusters between consecutive iterations).

With this idea of what k-Means do now we are going to discuss certain facts with respect to cluster behavior.

- The idea of clustering is to group data items having high similarity and to separate from dissimilar data items.
- The quality of a cluster is defined as high intra cluster similarity and low inter cluster similarity.

Having clustered our data, we now need some mechanism to choose test cases from each cluster. In the next section we used an algorithm called Pickup cluster that does the selection work.

PICKUP CLUSTER ALGORITHM:

K-Means algorithms clusters data items (test cases). That is the test cases in the same cluster have the same behavior. It would be redundant if we test different test cases from the same cluster because they would exhibit the same results. So in order to reduce this redundancy we need a selective approach of choosing test cases. This algorithm

proposes a method to choose tuple randomly from a cluster.

a)- Algorithm:

Input: Clustered data points from k-MEANS. Output: Single data point from each Cluster.

Step:

- For each cluster (1,….k), where each Ci contains all the tuples (ti1,ti2,….tin) that are mapped to it from k-Means.
- Pick one tuple randomly from (ti1,ti2,….tin).
- Add this tuple with the label Ci to file output.

At the end of execution of this algorithm we have one tuple from each cluster. In the next section we are going to see a detailed overview of our methodology which is based on the background provided on this section.

PROPOSED ALGORITHM:

(1) Generate test cases for the software using automation. Save them to a text file [19].

(2) Convert the file into attribute related file format (arff) according to the specifications in Weka.

(3) Load the converted arff file into Weka.

(4) Apply k-means algorithm to the above loaded data (k signifies how many clusters or in other words how many test cases we are looking for).

(5) Save the cluster assignments in an arff file.

(6) Take the clustered arff file and convert it into a simple text file by following the opposite process followed in step2

(7) Load the above text file into pickupCluster function .Execute the algorithm and save the output to a text file.

(8) Use this text file to test for coverage of the software.

(9) Repeat from step4 until acceptable coverage is achieved.

The complexity of K-Means algorithm is O(KNM) where K is the number of clusters, N is the number of test cases and M is the number of iterations. The complexity of Pickup cluster algorithm is O (K) where K is same as above.

CONCLUSION:

Our approach is not suitable to handle the large and complex system. This approach is very much suitable for simple systems where no more fork-joins, like nested-fork joins and etc. are involved, which is our next objective. However our proposed system is not sufficient to handle different kind of errors such as work flow errors, state based errors and etc.

REFERENCES:

[1] Abraham Silberschatz, Henry F. Korth and S. Sudarshan, “Database system concepts”, Internation Edition , 2006,pp 739-741.

[2] Ajitha Ranjan. “Automated Requirements-Based test case Generation”. Communications of ACM,2006

[3] Antonia Bertolino. “Software testing Research: Achievements, challenges and dreams” Future of Software Engineering, 2007.

[4] T. Y. Chen and M. F. Lau. A new heuristic for test suite reduction. Information and Software Technology, 40 (5):347-354, 1998.

[5] David Alex Lamb, “Software Engineering, planning for change,” Prentice Hall, Englewood Cliffs, NJ 07632, pp. 109– 112, 1988.

[6] M. J. Harrold, R. Gupta, and M. L. Soffa. A methodology for controlling the size of a test suit. ACM Trans. on Soft. Eng. and Meth., 2 (3):270-285,1993.

[7] A. E. Hassan, A. Mockus, R. C. Holt, and P. M. Johnson. Guest editor’s introduction: Special issue on mining software repositories. IEEE Trans. Softw. Eng., 31(6):426–428, 2005.

[8] A. K. Jain, M. N. Murty, and P. J. Flynn.A Data clustering: review. ACM Computing Surveys, 31(3):264–323, 1999.

[10] J. G. Lee and C. G. Chung. “An optimal representative set selection method”. Information and Software Technology, 42(1):17- 25, 2000.

[11] Lilly Ramesh, “Knowledge Mining of Test Case System,” International Journal on Computer Science and Engineering Vol.2(1), 2009, 69-73.

[12] Mark Last and Menahem Friedman. “The Data Mining approach to automated software testing.”. Communications of ACM,2003.

[13] Martina marre and Antonia Bertolino, “using spanning sets for coverage testing”. IEEE transactions on software Engineering, vol.29.

[14] Myra B Cohen and Matthew B Dwyer.”Coverage and adequacy in software product line testing”, Communications of ACM,2006.

[15] Remco R. Bouckaert, “Weka Manual 3-6-1”, Software manual, June 4,2009, pp-11-14.

[16] Tapas Kanugo and David M Mount.”A local search approximation algorithm for K-means clustering”. Communications of ACM,2002.

[17] Yanping Chen and Robert L Probert. “Regression test suite reduction using extended dependence analysis” Communications of ACM, 2007.