SEVERITY BASED CODE OPTIMIZATION : A DATA MINING APPROACH

M.V.P. Chandra Sekhara Rao

Department of CSE,R.V.R. &J.C. College of Engineering,ANU,GUNTUR.

Dr.B.Raveendra Babu

Department of CSE,R.V.R. &J.C. College of Engineering,ANU,GUNTUR

Dr. A.Damodaram

JNTU, CSE Department, JNTU College of Engineering, Kukatpally, Hyderabad,

Mrs.Aparna Chaparala

Department of CSE,R.V.R. &J.C. College of Engineering,ANU,GUNTUR

Abstract

Billions of lines of code are currently running in Legacy systems, mainly running machine critical systems. Large organizations and as well as small organizations extensively rely on IT infrastructure as the backbone. The dependability on legacy Software systems to meet current demanding requirements is a major challenge to any IT profession. One of the top priority of any IT manager is to maintain the existing legacy system and optimize modules where required. Various techniques have been developed to determine the complexity of the modules as well as protocols have developed to assess the severity of a software problem.

In this paper, it is proposed to study data mining algorithms in a multiclass scenario based on the severity of the error in the module.

Keywords— Legacy software, Normalization, Data mining, Random tree, Bayesian Logistic Regression, CART.

1.Introduction

Legacy systems are older software system and typically its original designers and implementers are no longer available to perform the system's maintenance. Often specifications and documentation for a legacy system are outdated / not available, so the only definitive source of information about the system is the code itself.

Organizations can have compelling reasons for keeping a legacy system, such as:

- The system is able to cope up with the current requirements and the management sees no need to change it.
- Cost of Retraining and lost time would be high.
- If the system is running 24/7 like in reservation ,it can not be taken out of service.
- A key problem in legacy system is the lack of documentation and the way the system works is not well understood.
- The user expects that the system can easily be replaced when this becomes necessary.

Frequently legacy systems are expensive to maintain and upgrade and have extreme limitations of function. They do not interface with new technologies well and available pool of support resources is dwindling. They are considered to be potentially problematic by many software engineers for several reasons Legacy systems often run on obsolete hardware, and spare parts for such computers may become increasingly difficult to obtain.

- The cost of maintaining the system will eventually outweigh the cost of replacing both the hardware and software
- Integration with newer systems may also be difficult because new software may use completely different technologies. The kind of bridge hardware and software that becomes available for different technologies that are popular at the same time are often not developed for differing technologies in different times, because of the lack of a large demand for it and the lack of associated reward of a large market economies of scale, though some of this "glue" does get developed by vendors and enthusiasts of particular legacy technologies

.

Where it is impossible to replace legacy systems through the practice of application retirement, it is still possible to enhance them. Most development often goes into improving the code of a legacy system.

Various computation methods have evolved with increasing number of lines of code which can give accurate .This data can

be utilized by extracting knowledge using Data Mining techniques.

Data mining becoming a very important tool already used in business intelligence, marketing intelligence, machine vision, genetic engineering, biotechnology and so on. Lot of research on data mining applicable to specific domains are being carried out by researchers in academic circles and in industries alike. For validation of algorithms and their effectiveness against known standards become extremely important. The need to compare results against a fixed a standard data set becomes important. Various organizations have assimilated huge repositories of datasets which can be used as a standard for validating algorithms and at the same time compare proposed systems with existing systems. In this research software reliability is focused, by proposing to use the NASA dataset and in particular the KC1 data set for classification and reliability prediction.

2.Classification principles in Data Mining

Classification is the process of finding a model (or function) that describes and distinguishes data classes or concepts for the purpose of being able to use the model to predict the class of objects whose class label is unknown. The derived model is based on the analysis of a set of training data. (i.e data objects whose class label is unknown). Classification predicts categorical (Discrete, unorder) labels prediction models continuous-valued functions. That is, it is used to predict missing or unavailable numerical data values rather than class labels. Regression analysis is a statistical methodology that is most often used for numerical prediction. Although other methods exist as well prediction also encompasses the identification of distribution trends based on the available data. Classification and prediction may need to be preceded by relevance analysis, which attempts to identify attributes that do not contribute to the classification or prediction process. This attributes can then be excluded.

It is proposed to study classification method in this work in order to classify the Software modules that need optimization. A System is developed, to classify the tuples into groups and to derive rules from these groups. A data mining system, equipped with several algorithms, identifies the software modules that need to be improved. The main aim of our data mining system is that it classifies the given input software modules and reduces the modules that are to be optimized.

3.Goal

In this work ,It is proposed to study data mining techniques and its application for classifying modules which require optimization and hence provide guide lines on software reliability. We carry on our work using available NASA dataset and in particular the KC1 data set for classification and reliability prediction. Classification is used to classify the software modules using software complexity measures like Halstead metrics on the KC1 (NASA) data set which is available in the public domain. We propose extensive study of methodologies which can be applied in tandem with existing classification systems to increase the accuracy of classification.

Severity of errors in a module can be classified as:

1. Catastrophic:

Defects that could (or did) cause disastrous consequences for the system in question.

2. Severe:

Defects that could (or did) cause very serious consequences for the system in question.

3. Major:

Defects that could (or did) cause significant consequences for the system in question - A defect that needs to be fixed but there is a workaround.

4. Minor:

Defects that could (or did) cause small or negligible consequences for the system in question. Easy to recover or workaround.

5.No Effect:

Trivial defects that can cause no negative consequences for the system in question. Such defects normally produce no erroneous outputs.

4.Data set

In the present work, the KC1 dataset available in the promise repository are used to generate models for defect classification. These datasets were collected from projects carried out at NASA's Metric Data Program (MDP) data repository and created under their metrics. Bonabeau, E., M. Dorigo et.al (2000) The datasets contain attributes composed of different LOC measure, cyclomatic complexity , Base Halstead Measures , Derived Halstead measures. For a comprehensive coverage and explanation of the metrics, referred vide Fenton and Pfleeger.

The attributes used in this work is described briefly below

LOC_BLANK - The number of blank lines in a module.

LOC_CODE_AND_COMMENT - The number of lines which contain both code & comment in a module.

LOC_COMMENTS - The number of lines of comments in a module.

CYCLOMATIC_COMPLEXITY - The cyclomatic complexity of a module.

DESIGN_COMPLEXITY - The design complexity of a module.

ESSENTIAL_COMPLEXITY - The essential complexity of a module.

LOC_EXECUTABLE - The number of lines of executable code for a module (not blank or comment)

HALSTEAD_CONTENT - The halstead length content of a module.

HALSTEAD_DIFFICULTY - The halstead difficulty metric of a module.
HALSTEAD_EFFORT - The halstead effort metric of a module.
HALSTEAD_ERROR_EST - The halstead error estimate metric of a module.
HALSTEAD_LENGTH - The halstead length metric of a module.
HALSTEAD_LEVEL - The halstead level metric of a module.
HALSTEAD_PROG_TIME - The halstead programming time metric of a module.
HALSTEAD_VOLUME - The halstead volume metric of a module.
NUM_OPERANDS - The number of operands contained in a module.
NUM_OPERATORS - The number of operators contained in a module.
NUM_UNIQUE_OPERANDS - The number of unique operands contained in a module.
NUM_UNIQUE_OPERATORS - The number of unique operators contained in a module.
LOC_TOTAL - The total number of lines for a given module.

It is proposed to use the Bayesian logistic regression classification algorithm and compare the result with Random tree classification algorithm and CART using weka.

Random trees are formed by a stochastic process. Random binary tree are binary trees with a given number of nodes, formed by inserting the nodes in a random order or by selecting all possible trees uniformly at random. Random trees can also be formed using spanning methods.

Classification and regression trees (CART) is a non-parametric technique that produces either classification or regression trees, depending on whether the dependent variable is categorical or numeric, respectively.

Trees are formed by a collection of rules based on values of certain variables in the modeling data set.

- Rules are selected based on how well splits based on variables' values can differentiate observations based on the dependent variable
- Once a rule is selected and splits a node into two, the same logic is applied to each "child" node.
- Splitting stops when CART detects no further gain can be made, or some pre-set stopping rules are met.
-

Each branch of the tree ends in a terminal node

- Each observation falls into one and exactly one terminal node
- Each terminal node is uniquely defined by a set of rules

6. Data Normalization

Databases are highly susceptible to noisy, missing, and inconsistent data due to their typically huge size (often several Giga bytes or more) and their likely origin from multiple, heterogeneous sources. Low-quality data will lead to low-quality mining results. There are number of data processing techniques. Data cleaning is one that can be applied to remove noise and collect inconsistencies in the data. Data integration merges data from multiple sources into a coherent data store, such as data warehouse. Data transformation, such as normalization, may be applied. Normalization may improve the accuracy and efficiency of mining algorithms involving distance measurement. Data reduction can reduce the data size by aggregating, eliminating redundant features, or clustering, for instance. These techniques are not mutually exclusive; they may work together. Data cleaning can involve transformations to correct wrong data, such as by transforming all entries for data field to a common format. Data processing techniques, when applied before mining, can substantially improve the overall quality of the pattern mined and/or not the time required the actual mining.

In this paper data is pre-processed using mathematical model i.e Normal Density Function.

- The equation for Normal Density Function (Cumulative = False) is given by f(x, μ, σ) = $1/\sqrt{2\Pi} \times \sigma * e^{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma}}$.

When cumulative is True, the Formulae is $\int_{-\infty}^{x} \frac{1}{\sqrt{2\Pi \times \sigma}} \times e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$
Normdist (x, mean, std, cumulative).

The meaning of the above function is as follows.

- x is the value of the attribute which we need to find the distribution.
- Mean is an arithmetic mean of the distribution.
- Std is the standard deviation of the distribution.
- Cumulative is a logical value which determines the form of the function.

+ If cumulative is time Normdist function returns cumulative distribution function, otherwise it returns the probability mass function.

The standard normal variate is given by

$$z = \frac{x - \mu}{\sigma}$$

For example from the given KC1 data set.

Halstead – content = 0 i.e., x = 0
Halstead – mean value 4.814858
Halstead – deviation is 3.713475

$$\therefore z = \frac{0 - 4.814858}{3.713475} = -1.296591$$

From normal distribution tables, the table value at 1.296591 is 0.0968.

If the cumulative is False, it returns the value of Normal density function is given by

$$\frac{1}{\sqrt{2\Pi \times \sigma}} \times e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$$= \frac{1}{\sqrt{2 \times \frac{22}{7} \times 3.713475}} \times e^{-\frac{1}{2}\left(\frac{0-4.814858}{3.713475}\right)^2}$$

$$= 0.0483$$

7. Experimental Investigation

In this section, our goal is to evaluate performance of the data mining classification techniques. Weka tool was used on KC1 data set for classification and the result is summarized in Table 2 and figure 1.

	% correctly classified	% Incorrectly Classified
Random Tree	94.5531	5.4469
Logistic regression	95.6704	4.3296
CART	96.7877	3.2123

Table 1. Percentage accuracy in classification

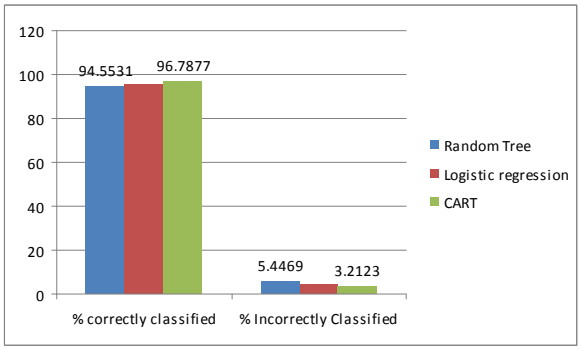

Figure 1. Bar chart of classification accuracy of Random tree, CART and Bayesian logistic Regression.

8.Conclusion

The Results tabulated in fig. 1shows that data mining is a viable method to predict modules that require optimization. The knowledge mined can be the starting point to the any IT manager to plan his strategy for software maintenance. The results can be validated for different data sets to establish uniformity of our proposed solution.

References

[1] Meacham, D.J.; Michael, J.B.; Man-Tak Shing; Voas, J.M, Standards interoperability: Applying software safety assurance standards to the evolution of legacy software, System of Systems Engineering, 2009. SoSE 2009. IEEE International Conference, Publication Year: 2009 , Page(s): 1 – 8.
[2] Hunold, S.; Korch, M.; Krellner, B.; Rauber, T.; Reichel, T.; Runger, G, Transformation of Legacy Software into Client/Server Applications through Pattern-Based Rearchitecturing Computer Software and Applications, 2008. COMPSAC '08. 32nd Annual IEEE International Digital Object Identifier:10.1109/COMPSAC.2008.158, Publication Year: 2008, Page(s): 303 – 310.
[3] Kangtae Kim; Hyungrok Kim; Woomok Kim; Building Software Product Line from the Legacy Systems "Experience in the Digital Audio and Video Domain", Software Product Line Conference, 2007. SPLC 2007. 11th International Digital Object Identifier: 10.1109/SPLINE.2007.27 Publication Year: 2007.
[4] Fayyad, U.M., Piatetsky-Shapiro, G., and Smyth, P., From Data Mining to Knowledge Discovery: An Overview. In Advances in Knowledge Discovery and Data Mining, Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P., and Uthurusamy, R. (eds.).
[5] Predicting fault prone modules by the Dempster-Shafer belief networks Guo, L. Cukic, B. Singh, H. Lane Dept. of CSEE, West Virginia Univ., Morgantown, WV, USA , 18th IEEE International Conference on Automated Software Engineering, 2003.
[6] Han J. and M. Kamber, Data Mining: Concepts and Techniques, 2nd edition. Morgan Kaufmann,
[7] A Machine Learning-Based Reliability Assessment Model for Critical Software Systems Challagulla, V.U.B. Computer Software and Applications Conference, 2007.
[8] Predicting Defective Software Components from Code Complexity Measures Hongyu Zhang; Xiuzhen Zhang; Ming Gu; PRDC 2007. 13th Pacific Rim International Symposium on Dependable Computing, 2007.
[9] A survey on metric of software complexity , Sheng Yu; Shijie Zhou; The 2nd IEEE International Conference on Information Management and Engineering (ICIME), 2010
[10] A Preliminary Performance Comparison of Two Feature Sets for Encrypted Traffic Classification Riyad Alshammari and A. Nur Zincir-Heywood Dalhousie University, Faculty of Computer Science
And Uthurusamy, R. (eds.), AAAI Press.