Journal of Advances and Scholarly Researches in Allied Education
Vol. XV, Issue No. 1, April-2018, ISSN 2230-7540

A Study of Software Reliability Classification Using Genetic Algorithm

Ranju Grover[1]* Dr. Y. P. Singh[2]

[1] Research Scholar of OPJS University, Churu, Rajasthan

[2] Associate Professor, OPJS University, Churu, Rajasthan

Abstract – As communicated previously, data mining is the path toward perceiving novel, possibly interesting and in the long run sensible precedents from significant volumes of data Piatetsky-Shapiro. Request is a basic thought in data mining. The gathering issue in data mining can be communicated as seeks after: given a couple of characteristics for data things and their related class names, envision the class stamp quality for data things for which it is dark. The potential employments of collection can be enormous. For example, portrayal may be used to discover the classes of customers who are most likely going to buy a thing and customers who won't. In like way, course of action can be used to arrange understudies as performers and under-performers with the objective that extended thought may be given to the understudies inclined to neglect to meet desires. Request can in like manner be used to uncover plans inclined to be found in phony trades.

Keywords: Software Reliability, Classification, Genetic Algorithm, Data Mining.

- X -

INTRODUCTION

This data consequently can be useful to recognize and square such false trades later on Macqueen et al.[58].According to Han and Chamber plan is extremely a two-advance process. In the underlying advance, the classifier attempts to pick up from a planning set of tuples whose class marks are known early. A tuple is addressed by a n- Dimensional characteristic vector d delineating estimations of Attributes. The tuple is expected to have a place with a class distinguished by the class mark Trait. The tuples used for getting ready and for whom the class stamp trademark is known early are called planning tuples or models. This system is properly depicted as "controlled learning". This movement can be viewed as the learning of a mapping limit y = f(X) where y is the class stamp and X is the trademark vector. The second step includes the estimation of the perceptive exactness of the classifier. Hence, a test set of tuples, routinely specific from the arrangement set is used. The precision is evaluated as the dimension of the test tuples successfully requested by the classifier. To choose this, the related class name trademark for each test tuple is separated and the classifier's measure. On the off chance that the accuracy is honorable, the classifier is set up to be utilized with already unnoticeable data. Han explain the capability among gathering and desire. The essential qualification is that with desire, as opposed to the class check property, a numeric measure which is a constant regarded limit is hoped to be settled.

Preprocessing: Han and Kamber portrayed the going with preprocessing steps that can unimaginably overhaul the precision of the classifier.

- Data Cleaning – incorporates removal or decline of noise and treatment of missing characteristics;
- Relevance examination – fuses relationship examination to decide if two attributes are quantifiably related. Similarly consolidates the dentification of unessential characteristics and quality subset decision to ascertain the most basic attributes as a response for thedimensionality issue;
- Data change and decline – may include institutionalization including the scaling everything considered so they fall inside a predefined range, and hypothesis including change of data to progressively raised sum thoughts

Evaluation of Classification Algorithms: The going with criteria are depicted by Han and Camber for assessing the execution of request figuring's

- Accuracy - the limit of the classifier to precisely envision the class sign of previously covered data;
- Speed - the count overhead realized by the classifier;
- Robustness - limit of the classifier to work with riotous data;
- Scalability - limit of the classifier to be used with high dimensional data;
- Interpretability - limit of the classifier to make justifiable principles.

Classification by Decision Tree Induction: Choice trees are a standout amongst the most well-known order strategies. A choice tree is a tree in which each non-leaf hub compares to a quality named as the part property and the branches speak to fundamentally unrelated

Result of a test on a part quality. The leaf hubs compare to classes. Figure 5.1 a case of choice tree is appeared as follows:

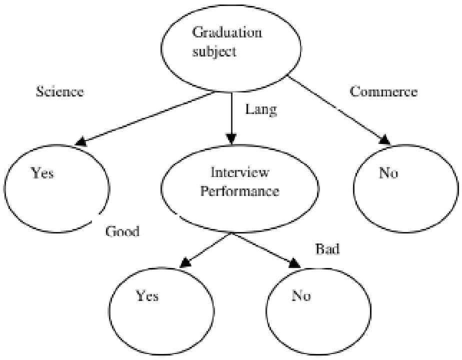

Figure 1- An Example Decision Tree

The major challenges to be addressed when working with decision trees include:

- Selection of the part trait – at any dimension of the choice trees, there must be a component to choose the part characteristic that yields the most ideal characterization at that dimension.
- Selection of the part measure – after the determination of the part quality, a decision should be made for the part foundation. In the event that the part quality is numeric, this can be a disparity. On the off chance that it is straight out, enrollment might be utilized.
- Over fitting – care should be taken that the choice tree developed does not speak to an "over fit" – one that performs extremely well with the preparation tuples yet not well with test tuples. This happens when the algorithm endeavors to take in a few idiosyncrasies that may exist in the preparation tuples yet which may not be available in the genuine dataset. To keep away from this, the choice tree built is exposed to a procedure called pruning which endeavors to evacuate at least one sub-trees supplanting them with leaf nodes. Various calculations exist in the writing for the development of choice trees. The calculations essentially contrast in the strategies used to choose the part characteristics. A portion of the measures utilized for deciding the part trait incorporate data gain, gain proportion and gini-list.

Information Gain:

The expected information needed to classify a tuple in D is expressed mathematically as

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

Where *pi* is the probability that an arbitrary tuples 1belongs to class C_i .

The information further needed in order to arrive at an accurate classification is described by the equation

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j)$$

Here, $Info_A$ (*D*) is the expected information required to classify a tuple from D

Based on partitioning by A. Information Gain is defined as

$$Gain(A) \ \square \ Info(D) - Info_A(D)$$

Gain Ratio

One of the problems with the information gain measure described above is that it tends to favor attributes with a large number of discrete values. To overcome the problem, the gain ratio measures take into account the split information measure and calculates the gain ratio as

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo(A)}$$

where gain is calculated as above. The formula for calculating the split information is given by:

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

Gini Index

The Gini-index measure focuses on maximizing the diversity. The equation for calculating the same is:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

Where p_i is the probability that a tuple in D belongs to class C_i .

Bayesian Classification: Factual procedures have the potential for grouping information tuples, and Bayesian classifiers abuse this thought. Bays Theorem expressed underneath frames the premise of Bayesian classification. Let X be an information tuple. It is depicted by n estimations. Give H a chance to be some theory, for instance, that an information tuple X has a place with a class C. P(H|X) is the likelihood that the speculation H holds given the tuple X. This is called back likelihood and should be resolved. P(H) is the earlier likelihood of H. For instance if P(H|X) means the likelihood that an understudy will fail to meet expectations in his graduation given his secondary school marks and monetary foundation. P(H) is the likelihood that an understudy will fail to meet expectations paying little respect to his secondary school marks and monetary foundation. P(X|H) is the back likelihood of X adapted on H. For the model, the likelihood an understudy has anchored under half in secondary school and is from a poor monetary foundation given that the understudy has failed to meet expectations. P(X) is the earlier likelihood of X. That is, the likelihood that a has anchored under half in secondary school and is from a poor financial foundation. Bayes Theorem expresses that:

$$P(H \mid X) = \frac{P(X \mid H)P(H)}{P(X)}$$

The Naïve Bayesian Classification

The Naïve Bayesian Classification follows the following steps:

Step 1 Let D be the training dataset where each tuple is represented by and tribte

$$X = (x_1, x_2, ..., x_n)$$

Step 2 Assuming there are m classes C_1 , C_2 ,..., C_m the classifier predicts that X belongsto the class having the highest posterior probability conditioned on X. X belongs to class C_i if and only if $P(C_i \mid X)$ $P(C_j \mid X)$ for $1 \leq j \leq m, j \neq i$

By Bayes Theorem, *P*

$$(C_i \mid X) = \frac{P(X \mid C_i)P(C_i)}{P(X)}$$

Step 3 As P(X) is constant for all classes, only the term $P(X \mid C_i)P(C_i)$ needs to be maximized.

Stage 4 If there are countless, calculation of P(X | Ci) may cause heinous overheads. The presumption of class contingent autonomy that the estimations of characteristics are restrictively free of each other - is utilized**.**

$$P(X \mid C_i) = \prod_{k=1}^{n} P(x_k \mid C_i) = P(x_1 \mid C_i) \times P(x_2 \mid C_i) \times ... \times P(x_n \mid C_i)$$

The computation of (|) P X Ci depends on whether the attributes are categorical or continuous. If attribute Ak is categorical, (|) k Ci P x is the number of tuples of class Ci in D having the value k x for An divided by the number of tuples of class Ci in D. If Ak is continuously valued, then (|) (, ,) Ci Ci k i k P x C = xg μ σ where g denotes the Gaussian distribution

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\Pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

Step *5* $P(X \mid C_i)P(C_i)$ is evaluated for each class C_i . X is considered to be ofclass C_i if and only if $P(X \mid C_i)$ P(C_i) □ $P(X \mid C_j)P(C_j)$ for $1 \leq j \leq m, j \neq i$

According to Han and Kamber[35], Bayesian classifiers theoretically yield the minimum error rate and have the potential of providing a theoretical justification for non-Bayes theorem-based classifiers.

Support Vector Machines

According to Noble a Support Vector Machine (SVM) is a computer algorithm that learns by example to assign labels to objects. Instances of applications of SVM include:

- Recognition of fraudulent credit card activity after examination of thousands of fraudulent and non-fraudulent credit card activity reports

- Recognition of handwritten digits
- Automatic classification of microarray gene expression profiles
- Classification of protein DNA sequences Underlying the operation of SVM are four basic concepts
- The separating hyper-plane
- The maximum-margin hyper-plane
- The soft margin
- The kernel function

GENETIC ALGORITHMS

Hereditary calculations have a place with the class of transformative calculation and are exceedingly suited for looking and advancement issues. These hereditary calculations try to discover answers for complex issues by copying the characteristic procedure of advancement. Part 3 gave an understanding into the working of hereditary algorithms. To utilize hereditary calculations for arrangement, arbitrarily produced principles frame the underlying populace. For instance, the standard "In the event that Blood Pressure > 160 and Blood Sugar > 300, Heart Failure=yes", can be encoded as 111. So also, the standard "In the event that Blood Pressure Not > 160 and Blood Sugar > 300, Heart Failure=no is encoded as 010. Another populace comprising of the fittest principles in the present populace is framed, and the development of this new populace considered posterity involves the utilization of administrators like traverse where substrings from a couple of rules are swapped to shape new guidelines and transformation where chosen bits are upset. The wellness of a populace is surveyed utilizing a wellness work that might be the classifier execution on the preparation tests. The way toward making new populaces proceeds until an ending measure is met.

SOFTWARE RELIABILITY

Designing of programming that can be trusted to give anticipated that administration should its clients has been a long standing test going up against programming professionals. The essential thought is to guarantee that the product conveyed to the client is solid before its conveyance. A great deal of barriers deflects the product build from accomplishing this. The principle issue is by all accounts the absence of sound strategies that guide in estimating the dependability of the product. Since what can't be estimated can't be controlled, building of solid programming is pressing problem facing the software community. The next few paragraphs are devoted to the understanding of the concept of reliability. Laprie defines “dependability” as the trustworthiness of a computer system so that reliance can justifiably be placed on the service it delivers. According to Somerville reliability is one aspect of dependability. The aspects of dependability as listed below:

- Availability – the likelihood that the framework will be ready for action and ready to convey valuable administrations at some random time;
- Reliability – the likelihood over a given timeframe that the framework will effectively convey benefits not surprisingly by the client;
- Safety – the probability that the framework will make harm individuals or its condition;
- Security – the probability that the framework can oppose unintentional or intentional interruption. This research focuses on reliability. The IEEE definition of software reliability is given as “the ability of a system or component to perform its required functions under stated conditions for a specified period of time”.

SOFTWARE RELIABILITY CLASSIFICATION USING GENETICALGORITHM

This research attempts to capitalize on the power of genetic algorithms in uncovering accurate classification models. The problem can be stated as below – “Given a set of characteristics of a software module (metrics), classify it as reliable or unreliable”. Such identification can alert the software project manager to the need of extensive testing of the modules predicted to be “unreliable”. This in turn can reveal faults which can be corrected before the software is delivered.

Metrics considered for the study: A module is spoken to by a lot of measurements. These measurements will be estimations of specific properties of the module under thought. Calculation of these measurements esteems should be possible physically, or many mechanized instruments are accessible for their calculation. A module is represented by a set of metrics. These metrics are measurements of certain properties of the module under consideration. Computation of these metrics values can be done manually, or many automated tools are available for their computation.

Experimental Setup: So as to construct a hereditary calculation based classifier, initial 213 modules created by a nearby programming association are utilized as preparing tests. The unwavering quality classes of these modules are known. For the motivations behind the test, a module with at least 5 detailed number of deficiencies in the initial three months was considered as "inconsistent" and that with under 5 issues was considered "dependable". Out of the 213 chose modules, 181 were in the dependable Class and 32 were in the "untrustworthy" classification. The qualities for every one of the measurements recorded in Table 5.2 are known for the 213 modules. So as to apply hereditary calculation for the issue, arrangements are spoken to by chromosomes containing qualities that are the estimations of the measurements. The wellness work is assessed by utilizing the Linear Discriminant Analysis (LDA) – which is a classifier procedure expressed by Duda and utilized by Valance and Pizzi with the forget one technique for preparing and testing. This implies initial a module is chosen and preparing is finished with all the staying 212 modules, and it is seen whether they chose module is effectively arranged. The procedure is rehashed for all the 212 modules. The quantity of chromosomes in a populace was set at 200, and the quantity of first class qualities is settled at 50. This implies in every age after the chromosomes are arranged in diminishing request of wellness, the main 50 chromosomes are passed on to the cutting edge as "world class" qualities. An irregular likelihood P is created and from the staying 150 qualities, 2 qualities with wellness more prominent or equivalent to P are chosen as parent qualities. A traverse point is chosen aimlessly, and another posterity is delivered by joining bits from both the guardians. On the off chance that the produced likelihood is more noteworthy than 1 – 10%=0.9, the posterity is transformed by changing each piece from 0 to 1 and the other way around. The procedure of traverse and change are rehashed until the point when 150 new offspring's are made, and with the new populace of the 50 first class qualities in addition to the 150 offspring's, the entire procedure is rehashed for 200generations. The order rate was the occasions the module forgotten was accurately gathered.

CONCLUSION

Although, BP used the validation data for training and trained for many more epochs, it still performed poorer than the GA, for these classification problems. A critique of this study might be that the BP configuration used is limited to one set of user defined parameter settings. By changing any or all of the parameters available to users of the BP algorithm could possibly result in better performance. However, since there are no heuristics for setting these parameters and each BP NN is problem dependent, the problem facing the decision maker is which set to choose and recommendations provided in the software are reasonable choices. The GA, as used in this study on the other hand, has predefined parameter settings and can be readily used for any given problem. Our findings indicate that the GA is not dependent upon the initial random weights for finding superior solutions and is more consistent and predictable in its abilities for finding those solutions. This research has shown the GA to be a viable alternative for NN optimization that finds superior solutions in a more consistent and predictable manner than those using BP. The GA may enable managers to use neural networks with more confidence that the reported solution is in fact the desired solution and thus allow the NN to become a powerful tool for managers.

REFERENCES:

1. Karegowda A.G., Vidya T., Shama M. Jayaram A. Manjunath A. S. (2012). "Improving Performance of K-Means Clustering by Initializing Cluster Centers Using Genetic Algorithm and Entropy Based Fuzzy Clustering for Categorization of Diabetic Patients" Proceedings of International Conference on Advances in Computing Advances in Intelligent Systems and Computing Volume 174, pp. 899-904.

2. Lu B. and Ju F. (2012). "An optimized genetic K-means clustering algorithm", IEEE International Conference on Computer Science and Information Processing (CSIP), pp. 1296–1299.

3. Padhy N., Dr. Mishra P. and Panigrahi R. (2012). "The Survey of Data Mining Applications and Feature Scope". International Journal of Computer Science Engineering and Information Technology (IJCSEIT), Vol. 2, No. 3.

4. Peter P. and Baryamureeba W.V. (2008). "Extraction of Interesting Association Rules Using Genetic Algorithms", International Journal of Computing and ICT Research, Vol. 2, No. 1, pp. 26-33.

5. Sarafis I., Zalzala A.M.S. & Trinder P.W. (2002). "A Genetic Rule-Based Data Clustering Tool Kit", Conference on Evolutionary Computation (CEC), Honolulu, USA.

6. Sharma's and Shikha Rai (2012). "Genetic K-Means Algorithm – Implementation and analysis" International Journal of Recent Technology and Engineering (IJRTE),ISSN: 2277-3878, Volume-1, Issue-2.

7. Singh P.D., Jian N. and Vidisha S.A.T.I. (2013). "Improved Partition Clustering Algorithm (k-means) Based on Genetics" International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume 2 Issue 2 Page No. 499-504.

8. Vivanco R. and Pizzi N. (2004). "Finding Effective Software Metrics to Classify Maintainability Using a Parallel Genetic Algorithm", Proceedings Part II of Genetic

and Evolutionary Computation – GECCO 2004.

9. Wahidah H., Low P.V., Ng L.K. and Ong Z. L. (2011). ”Application of Data Mining Techniques for Improving Software Engineering”, School of Computer Sciences: Publications.

10. Wakabi-Waiswa P.P. and Baryamureeba V. (2008). ”Extraction of Interesting Association Rules Using Genetic Algorithms”, International Journal of Computing and ICT Research, Vol. 2, No. 1, pp. 26-33.

11. Yan X., Zhang C. and Zhang S. (2009). “Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support”, Expert systems with Applications Vol. 36, pp. 3066–3076.

Corresponding Author

Ranju Grover*

Research Scholar of OPJS University, Churu, Rajasthan