
A. Shanthini¹, G.Vinodhini² and RM. Chandrasekaran³

¹Research Scholar, Department of Computer Science and Engineering, Annamalai University, Annamalai nagar, Tamil Nadu, India.
²Assistant Professor, Department of Computer Science and Engineering, Annamalai University, Annamalai nagar, Tamil Nadu, India.
³Professor, Department of Computer Science and Engineering, Annamalai University, Annamalai nagar, Tamil Nadu, India.

ABSTRACT: Machine Learning (ML) approaches have a great impact in fault prediction. Demand for producing quality assured software in an organization has been rapidly increased during the last few years. This leads to increase in development of machine learning algorithms for analyzing and classifying the data sets, which can be used in constructing models for predicting the important quality attributes such as fault proneness. Defective modules in software project have a considerable risk which reduces the quality of the software. This paper mainly addresses the software fault prediction using hybrid Support Vector Machine (SVM) classifier. We conduct a comparative study using the WEKA tool for three different levels of software metrics (package level, class level and method level) with hybrid SVM classifiers using feature selection techniques such as Principle Component Analysis (PCA). The experiments are carried out on the datasets such as NASA KC1 method level data set, NASA KC1 class level dataset and Eclipse dataset for package level metrics. The feature selection techniques evolved by experiments shows that Principle Component Analysis (PCA) with hybrid SVM performs better than other feature selection techniques.

Keywords - Fault prediction, Machine Learning, Feature Selection techniques.

1. INTRODUCTION

Software fault proneness is a vital factor in analyzing the quality of the software product. The fault proneness of a class predicts the probability of the presence of faults in the module. A model for prediction of software fault proneness allows software organizations to identify faulty modules in the software. The quality of the software can be analyzed in the early stages of the software development life cycle. The values of the metrics found in the predicted model are used for assessing the software quality. These quality models can be used for the system under development and maintenance. As the complexity and constraint in the developing software is increasing, it is difficult to produce software without any faults. Effective fault prediction model can aid the developers to focus on quality assurance activities on fault prone models [2]. These models often use static measures obtained from source code. They mainly concentrate on size, coupling, cohesion, inheritance and complexity measure which have been associated with risk factor, such as fault and changes. Faulty modules in the product cause considerable risk by decreasing customer satisfaction and by increasing the development and maintenance costs. Software fault prediction using machine learning algorithms are investigated by many researchers. Many new literatures reviews show the performance difficulty by applying a single classifier. On the other hand, classifiers ensemble can effectively improve classification performance than a single classifier [13]. The above reasons indicate that fault prediction using classifiers ensemble methods have not been fully exploited. We conduct a comparative study of bagged SVM ensemble using feature selection methods (PCA, Random projection, Random Subset) on different data level metrics with perspective of taxonomy. This study is focused on the software fault prediction based on Machine Learning (ML) algorithms. The use of machine learning algorithms have proven to be of great practical value in solving a variety of engineering problems including the prediction of failure, fault, and fault-proneness. One of the most active areas of recent research in machine learning has been the use of ensemble classifiers.
The primary objective of this research is to predict the faults in a module using machine learning algorithms and analyze the feature selection technique to show that Principal Component Analysis (PCA) has better performance than other feature selection techniques (Random Subset and Random Projection) while using hybrid SVM classifier.

2. REVIEW OF LITERATURE

Considerable research has been performed on software metrics and fault prediction models. Catal [3] examined Chidamber-Kemerer (CK) metrics suite and some method-level metrics (the McCabe’s and Halstead’s ones) for a fault model. This fault module is based on Artificial Immune Recognition System (AIRS) algorithm. The authors investigated 10 metrics from the class-level metrics and 84 metrics from the method-level metrics. According to the obtained results the authors concluded that the best fault prediction is achieved when CK metrics is used rather than McCabe’s and Halstead’s metrics.

Menzies, DiStefano, Orrego, and Chapman [1] applied the machine learning algorithms for fault prediction. Naive Bayes algorithm is one of the efficient machine learning algorithms used in fault prediction. The data set used for this research is public datasets locating in PROMISE repository. Navies Bayes is applied in Method level metrics for fault prediction. The performance evaluation metrics used for analysis are probability of detection (PD) and probability of false alarm (PF). When comparison is made, Naive Bayes provides better performance than J48 classification algorithm. Furthermore, they reported that PD on KC1 dataset is 55% and PD for Fagan inspections is between 33% and 65%. For industrial inspections, PD for Fagan inspections is between 13% and 30%.

Kanmani [7] validated Probabilistic Neural Network (PNN) and Back propagation Neural Network (BPN) using a dataset collected from projects of graduate students. The results obtained were validated for performance using statistical techniques. According to Kanmani [6] study, PNN provided better performance than BPN. Object oriented metrics is evaluated by Amjan Shaik [11].

Catal, Ugur and Diri[12] developed an Eclipse-based software fault prediction tool for Java programs in order to simplify the fault prediction process. They also integrated a machine learning algorithm called Naive Bayes into this plug-in because of its proven high-performance for this problem. As this tool proved the machine learning algorithms can be easily adopted for real world applications. They also examined the probabilistic results of the Naive Bayes implementation.

The use of Machine Learning algorithms for the purpose of predicting the fault prone modules in a software is proposed by Gondra [10], which views fault-proneness as both a continuous measure and a binary classification task. Using a NASA public dataset, Artificial Neural Network (ANN) was used to predict the continuous measure while a SVM was used for the classification task.

3. FAULT PREDICTION DATA SETS

Three different data sets used in this research are obtained from the bug database of Eclipse and NASA IV and V Facility Metrics Data Program (MDP). All data are publicly available and used for fault prediction models. For method level metrics, NASA KC1 method level dataset is used which include 21 method-level metrics proposed by Halstead and McCabe [3]. This dataset consists of 2109 instances and 22 attributes [8].

We use the data associated with the KC1 project for class level metrics. This is a real time project written in C++ consisting of approximately 315,000 lines of code (LOC). There are 10,878 modules and 145 instances. For package level metrics, the dataset is obtained from bug database of Eclipse 3.0. The dataset lists the number of pre- and post-release faults for every package in the Eclipse 3.0.
4. FEATURE SELECTION TECHNIQUES

Feature selection is a technique used to reduce the number of features (metrics) in a dataset that are irrelevant and cause negative effects on the prediction task. Feature selection technique for a dataset is applied before applying a data mining algorithm. For feature selection in unsupervised learning, learning algorithm are designed in such a way that it finds a natural grouping of relevant features in a data set that helps in prediction task. Thus feature selection in unsupervised learning aims to find a good subset of features for giving optimized result in prediction task. There are various feature selection techniques of which we are using PCA, Random projection and Random subset for analysis.

4.1 Principal Component Analysis

Principal Components Analysis (PCA) is the widely used statistical method to reduce the effect of these correlations. Principal components analysis can transform the original set of correlated variables into a smaller set of uncorrelated variables that are linear combinations of the original ones for most robust modeling.

4.2 Random Projection

Random projections are a method of dimensionality reduction. Random projections involve taking a high-dimensional data-set and then mapping it into a lower-dimensional space. With dimensionality reduction, it should be clear that whether a data-set needs all its dimensions.

4.3. Random Subset

Random subset selects a random subset of attributes. For example, in order to create a subsets of the training data with 5, 10, 15, and 24 elements, we have to create 3 subsets of each size. In Random subset algorithm, the algorithm picks a random subset of size N from the integers [1...X], where, \( N \leq X \).

5. HYBRID SVM CLASSIFIER

Ensemble learning techniques have been shown to increase machine learning accuracy by combining arrays of specialized learners. Bagging and boosting [4, 9] are examples of ensemble methods. Bagging is a “bootstrap” ensemble method that creates individuals for its ensemble by training each classifier on a random redistribution of the training set.

5.1. Bagging

Bagging is Bootstrap AGGregatING. The main idea of Bagging is to construct each member of the ensemble from a different training dataset, and to predict the combination by either uniform averaging or voting over class labels [1]. A bootstrap samples N items uniformly at random with replacement. That means each classifier is trained on a sample of examples taken with a replacement from the training set, and each sample size is equal to the size of the original training set. Therefore, Bagging produces a combined model that often performs better than the single model built from the original single training set.

6. EXPERIMENTS

In this research paper, the WEKA machine learning library is used as the source of algorithms for experimentation. The bagging and SVM classification algorithms are implemented in WEKA with default parameters.

6.1 Root Mean Square Error Rate

RMSE is frequently used measure of differences between values predicted by a model or estimator and the values actually observed from the thing being modeled or estimated. It is just the square root of the mean square error as shown in equation given below. Assuming that the actual output is \( a \), expected output is \( c \).
To evaluate the performance of three feature selection techniques, SVM classifier is used. The data set described in section 2 is being used to test the performance of Support Vector Machine. Root Mean Square Error (RMSE) was evaluated using 10-fold cross validation as cross validation is the best technique to get a reliable error estimate. The Root Mean square error in the figure 1 reflects the best performance of random subset with SVM in terms classification rate. The data set is being used to test the performance of Hybrid Support Vector Machine. It is clearly evident from figure 2 that, PCA with Hybrid SVM performs better than other feature selection techniques for various levels of software metrics. The performance of Random subset is also approximately equivalent to P.

\[
\sqrt{\frac{1}{n} \sum_{i=1}^{n} (a_i - c_i)^2 + (a_2 - c_2)^2 + ... + (a_n - c_n)^2}
\]

6.2 Area under Curve (AUC - ROC)

The dataset described in section 2 is being used to test the performance of various ensemble methods. We adapted ROC curve in our experiment to evaluate the performance of hybrid algorithm. Receiver operative Characteristics (ROC) curve is used as an additional alternative evaluation metric. AUC-ROC is used as a performance metrics (area under ROC curve), an integral of ROC curve with false positive rate as x axis and true positive rate as y axis. The evaluation is done using AUC-ROC for SVM and Hybrid SVM using various feature selection techniques.

<table>
<thead>
<tr>
<th>FEATURE SELECTION METHODS</th>
<th>AREA UNDER ROC CURVE USING SVM</th>
<th>AREA UNDER ROC CURVE USING HYBRID SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>METHOD LEVEL METRICS</td>
<td>CLASS LEVEL METRICS</td>
</tr>
<tr>
<td>PCA</td>
<td>0.58</td>
<td>0.763</td>
</tr>
<tr>
<td>Random Projection</td>
<td>0.503</td>
<td>0.606</td>
</tr>
<tr>
<td>Random Subset</td>
<td>0.502</td>
<td>0.774</td>
</tr>
</tbody>
</table>

Figure 3, 4 and 5 shows the ROC curves evaluating the performance curve of SVM classifier and figures 6, 7 and 8 depicts the ROC curves evaluating the performance curve of Hybrid SVM classifier on the KC1 method level dataset, KC1 class level dataset and Eclipse package level dataset. Area under the ROC curve (AUC-ROC)
is calculated using trapezoidal method (as given in Table1) using various feature selection techniques for SVM and Hybrid SVM. The values tabulated in the Table1 shows that PCA with Hybrid SVM performs better than individual SVM classifier for different levels of metrics using various feature selection techniques. From the ROC curves (AUC-ROC) it is evident that, for method level metrics, class level and package level metrics PCA gives better performance among other feature selection techniques for different level of metrics.

Figure 3: ROC for method level metrics using SVM

Figure 4: ROC for class level metrics using SVM

Figure 5: ROC for package level metrics using SVM

Figure 6: ROC for method level using Hybrid SVM

Figure 7: ROC for class level metrics using Hybrid SVM

Figure 8: ROC for package level metrics using Hybrid SVM

7. CONCLUSION

The goal of our research is to analyze the performance of various feature selection techniques using bagged SVM classifier for various metrics level data set on fault prediction. We analyzed the performance of the filters (PCA, Random projection, Random Subset) using Root Mean Square Error. ROC is also used as an alternative metric. The area under ROC curves (AUC-ROC) is calculated by using the trapezoidal method. From the ROC curve (AUC-ROC) it is evident that, for three different metrics level dataset (KC1 method level dataset, KC1 class level dataset and Eclipse dataset) hybrid SVM gives better performance in terms of classification rate. The feature selection technique evolved shows that PCA with hybrid SVM performs better than other feature
selection techniques. When AUC is used as an evaluation metric, for KC1 class level dataset and Eclipse package level dataset, fault prediction rate is 84% and 78% with PCA which is better when compared with other feature selection methods. For KC1 class level dataset, fault prediction rate is 84% with hybrid SVM using AUC-ROC as a metric. Hybrid SVM using PCA outperforms other feature selection methods when performance is evaluated using both RMSE and AUC-ROC. We plan to replicate our study to predict the models based on other machine learning algorithms such as ensemble using neural networks and genetic algorithms.

8. REFERENCES


[8] Dr. Kadhim M. Breesam., 2007, Metrics for Object Oriented design focusing on class Inheritance metrics, 2nd International conference on dependability of computer system IEEE.


