Software Defect Prediction via Convolutional Neural Network

To improve software reliability, software defect prediction is utilized to assist developers in finding potential bugs and allocating their testing efforts. Traditional defect prediction studies mainly focus on designing hand-crafted features, which are input into machine learning classifiers to identify defective code. However, these hand-crafted features often fail to capture the semantic and structural information of programs. Such information is important in modeling program functionality and can lead to more accurate defect prediction.

Software defect prediction is a process of building classifiers to predict code areas that potentially contain defects, using information such as code complexity and change history. The prediction results (i.e., buggy code areas) can place warnings for code reviewers and allocate their efforts. The code areas could be files, changes or methods.

Related Work

There are many studies about software bug prediction using machine learning techniques. The study predicts the software future faults depending on the historical data of the software accumulated faults. The study also evaluated and compared the AR model and with the Known power model (POWM) used Root Mean Square Error (RMSE) measure. In addition to, the study used three datasets for evaluation and the results were promising.

Sharma and Chandra added to their study the most important previous researches about each ML techniques and the current trends in software bug prediction using machine learning. This study can be used as ground or step to prepare for future work in software bug prediction.

D. L. Gupta and K. Saxena developed a model for object-oriented Software Bug Prediction System (SBPS). The study combined similar types of defect datasets which are available at Promise Software Engineering Repository. The study evaluated the proposed model by using the performance measure (accuracy). Finally, the study results showed that the average proposed model accuracy is 76.27%.

Bavisi presented the most popular data mining technique (k-Nearest Neighbors, Naïve Bayes, C-4.5 and Decision trees). The study analyzed and compared four algorithms and discussed the advantages and disadvantages of each algorithm. The results of the study showed that there were different factors affecting the accuracy of each technique; such as the nature of the problem, the used dataset and its performance matrix.

Problem Statement

Software bug prediction is a technique in which a prediction model is created in order to predict the future software faults based on historical data. Various approaches have been proposed using different datasets, different metrics and different performance measures. But Problem is Accuracy. Accuracy is less then 90%.

Our Objective

1. First objective is to improve the Accuracy more then 90%.
2. Adding more software metrics in the learning process is one possible approach to increase the accuracy of the prediction model.
3. Implement C4.5 Algorithm with ANN and CNN.