CHAPTER 6

SUMMARY AND CONCLUSION

In the modern era many object-oriented software, quality suites are increasingly used for assessing quality against features of object-oriented design and as well as against the factors of evaluating quality. Today, the contribution of metrics to the overall objective of software quality is understood and fully recognized by the software engineering community. Businesses have learned that measuring the effectiveness of software projects can impact the bottom line and quality is no longer an advantage in the software marketplace but it is a necessity. For these reasons, the demand for quality in software engineering has taken center stage in the twenty-first century.

The main objective of this research work is to design and develop fault prediction models using machine learning classifiers and object oriented software quality metrics to predict fault proneness in modules and also to extend the existing quality metrics to include object oriented characteristics like friend function and inheritance.

The study analyzed four categories of existing metrics, namely, simple metrics, Chidamber and Kemerer (CK), Metrics for Object Oriented Design (MOOD) and program complexity metrics. Six simple metrics were considered. They are, LOC (Total number of lines), BR (Number of methods), NOP (Total Number of Unique Operators), NOPE (Total Number of Unique Operands), RE (Readability with Comment percentage) and VO (Volume). The CK metrics suite consists of Weighted Methods Per Class (WMC), Depth of Inheritance Tree (DIT), Number of Children (NOC), Coupling Between Objects (CBO), Response For a Class (RFC), Lack of Cohesion in Methods (LCOM). The MOOD metrics considered are Method Hiding Factor (MHF), Attribute Hiding Factor (AHF),
Method Inheritance Factor (MIF), Attribute Inheritance Factor (AIF), Polymorphism Factor (POF) and Coupling Factor (COF). The program complexity metrics estimated are Cyclomatic complexity metric and Fan-In Fan-Out Complexity metric. These metrics are termed as EQMF metric set in this study.

Apart from the above existing metrics, the study proposes the use of four extended object oriented metrics, namely, Class Method Flow Complexity Measure (CMFCM), Friend Class Complexity Metric (FCCM), Class Complexity from Inheritance (CCI) and Class Complexity from Cohesion Measure (CCCM). These metrics were proposed to calculate module complexity by considering characteristics that are not dealt by the existing metrics. This metric set is termed as ExQMF in this study.

A genetic algorithm based fusion and selection algorithm is proposed to select significant features and combine EQMF and ExQMF sets to form a fused metric set. This set is termed as FQMF metric set in this study. A Sensitivity Casual Index (SCI) is proposed as dimensionality reduction algorithm for solving the problem of ‘curse of dimensionality’.

During fault proneness prediction, the applicability of Ensemble of Classifiers (EoC) is proposed. For this purpose, three classifiers, namely, Back Propagation Neural Network (BPNN), Support Vector Machine (SVM) and K-Nearest Neighbour (KNN) were used. These classifiers were selected because of its high accuracy and high diversity properties. Using these three classifiers, both homogeneous and heterogeneous EoC were designed and build. Four sub-sample selection algorithms, namely, Bagging, Boosting, Sequential Selection and Random Subspace Selection were used during the construction of EoC Systems. Hold-Out method (H-Method) was used to split the metric set into training and testing. A modified majority voting scheme that combines weighting scheme is used as the aggregate method. Twelve homogeneous systems and 16
heterogeneous systems were built to EoC predict proneness in object oriented software.

The proposed fault prediction systems were evaluated using four projects from which the selected metrics were estimated. Five performance measures, namely, accuracy, precision, recall, F-Measure and speed of prediction were used to analyze the prediction systems. The experimental results showed that the SVM with bagging method produced best results while using fused metric set (FQMF) among homogeneous systems and while comparing the heterogeneous systems, BSKBA (3-classifier heterogeneous system using bagging method) produced improved results while using FQMF.

In any software project, there can be a number of faults. It is very essential to deal with these faults and to try to detect them as early as possible in the lifecycle of the project development. In this study, it is dealt by using an existing and extended software quality metrics and homogeneous, heterogeneous ensemble systems. The various experimental results shows that the proposed systems with fused metric set are successful in predicting faults in object oriented software.

6.1 FUTURE RESEARCH DIRECTIONS

The following points can be considered in future to further improve the performance of 2OFSP systems.

- A dynamic ensemble selection model instead of a dynamic classification selection algorithm can be designed to efficiently select different heterogeneous Ensemble of Classifiers.

- A preprocessing routine to handle outliers and missing values can be included to further improve the performance of the EoC.

- Classifier pruning algorithms can be incorporated to handle the increased time complexity of the EoC.