A Hybrid Approach of Module Sequence Generation using Neural Network for Software Architecture

Manjot Kalsi¹, Janpreet Singh²

¹M.Tech Research Scholar, Lovely Professional University, Phagwara, Punjab, India
²Assistant Professor, Department of CSE, Lovely Professional University, Phagwara, Punjab, India

Abstract: Software Engineering Principals gave a new direction to software industry [7]. With the advent of software engineering principals we are now able to check the feasibility of modules under consideration which further will lead to restrict the happening of software crisis [9]. But the selection of development of modules from the bunch of modules identified is totally dependent on the past experience of the project manager or project planner. For the detection of fault prone modules which will lead to software crisis if we are not implementing then in a desired order, various predictive models can be used based on source code metrics as input for the classifiers [13]. The identification of metric subset for the enhancement of performance for predictive objective would not only improve the model but also provides strength to the structural properties of the modules. Unbalanced datasets also itself is a kind of difficulty for building predictive modeling. Unbalanced datasets are common in empirical software engineering as a majority of the modules are not likely to be faulty [8, 28]. We propose a method of applying search based metric selection and oversampling of NASA dataset [3]. This paper is the extension of our previous review paper. In this proposed method the selection approach uses the weights of Neural Network to identify the sequence of implementation of software modules.

Keywords: Metric subset, Predictive models, unbalanced datasets, NASA dataset, Neural Network.

1. Introduction

Many software organizations adapt the solutions provided by different software life cycle models. The refining of different steps to achieve final objective by these life cycle models helps to produce quality products as well as helps to understand in advance about the feasibility or acceptance of final work done by the organization. But the main aim of organization is to produce high quality products in a timely manner as well as cost effective manner [1]. To increase the utilization of available resources, when taking mitigating actions, such as code inspection, refactoring etc., the ability to identify potentially, referenced components would assisted. Predictive models have been a focus of research in empirical studies of software systems [20]. Life cycle models fulfill the quality criteria but along with the quality criteria we have to deliver or develop the products on timely bases. So at each and every step we have to improve or enhance the criteria of selection of steps of development. Number of researchers work on this area of quantification of implementation and structural design of systems [4, 30]. Their study observes that feature selection is the process of identifying a subset of features that improves a model’s specific performance [5]. Many researchers suggest to use search based optimization with testing release date planning and cost estimation [31, 32]. In this study we propose a Neural Network based algorithm as a search based feature selection strategy in order to find a subset of source code metrics that will generate an implementation sequence that enhances and simplifies a predictive model of software quality [6, 10]. To built predictive models for generating efficient sequence of modules for implementation, fit databases are required which contains instances of faulty and non-faulty modules. Preparation of balanced fit dataset is also always not possible when using industrial systems. Many modules in the fit dataset are actually imbalanced i.e. these exists a large difference between the number of fault prone modules and non faulty modules. This causes performance degradation of fault-proneness modules [15, 19].

In conventional systems, the predictive accuracy of minority class is degraded since the prediction accuracy of the majority class is dominant in satisfying the objective function of model [25, 2]. In this paper we propose sampling method to balance the fit dataset in order to increase classifier performance as a classifier we took cohesion and coupling parameters of different modules under consideration. In our previous paper we proposed this technique while doing review in this field and this paper contains the result part obtained after the implementation of proposed idea. This paper shows the performance of Artificial Neural network in accordance with the proposed system. In the following sections we discussed the objectives of over work then the methodology used by us to complete the task, and then the next section contains the information about the NASA dataset KC1used to observe the working of the proposed system. At the end we have the result part in the paper. This result part shows the performance of the proposed system with the help of parameters like Mean Absolute Error, Root Mean Square Error, and Accuracy etc.

2. Proposed Objectives

The different objectives to achieve the successful implementation are discussed below. These steps are used to find the sequence of execution of modules present in the system. This sequence will produce efficient results as per the conclusion made on the bases of parameters of comparison.

- Selection of modules under consideration from KC1 dataset.
To find the maximum value of cohesion and coupling parameters of selected modules to find the percentage of cohesion and coupling for each selected module.

- To find the collective percentage of cohesion and coupling for a single module.
- To assign weights to the nodes of Neural Network by the values of collective percentage calculated for each module.
- To find sequence of implementation of modules on the basis of weights assigned to the Neural Network as the final output of the system.
- To find the performance measures of the modules sequence as SNR, RMSE, CPU time and accuracy.

3. Methodology

The following diagram describes the flow of methodology used in the proposed system. Some of the features of the software are selected to find the sequence of implementation of modules. In the proposed system we use coupling and cohesion as parameters to take decision about the generation of module sequence. Neural Network helps to identify the exact module to be implemented as we are considering the percentage of different parameters to take decision about which module to implement next. The judgment is done on the basis of weights assigned to the nodes of the Neural Network.

![Proposed Methodology Diagram](image)

### Table 1: KC1 source code metrics

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>loc blank</td>
<td>branch count</td>
</tr>
<tr>
<td>loc code and comments</td>
<td>loc comments</td>
</tr>
<tr>
<td>cyclomatic complexity</td>
<td>design complexity</td>
</tr>
<tr>
<td>essential complexity</td>
<td>loc executable</td>
</tr>
<tr>
<td>halstead content</td>
<td>halstead difficulty</td>
</tr>
<tr>
<td>halstead effort</td>
<td>halstead error est</td>
</tr>
<tr>
<td>halstead level</td>
<td>halstead length</td>
</tr>
<tr>
<td>halstead prog time</td>
<td>halstead volume</td>
</tr>
<tr>
<td>num operands</td>
<td>num operators</td>
</tr>
<tr>
<td>num unique operands</td>
<td>num unique operators</td>
</tr>
<tr>
<td>loc total</td>
<td></td>
</tr>
</tbody>
</table>

5. Artificial Neural Network

An Artificial Neural Network (ANN) is an information-processing paradigm that is inspired by the way a biological nervous system in human brain works [11, 18]. Large number of neurons present in the human brain forms the key element of the neural network paradigm and act as elementary processing elements. These neurons are highly interconnected and work in unison to solve complex problems. Likewise, an Artificial Neural Network can be configured to solve a number of difficult and complex problems. ANNs find a wide variety of applications in diverse areas including functional approximation, nonlinear system identification and control, pattern recognition and pattern classification, optimization, English text pronunciation, protein secondary structure prediction and speech recognition [12, 29]. Fundamentals of ANNs and its salient features are discussed in the following sections.

6. Results

In this results section we are considering the data obtained from different runs of the system by considering different number of modules each time. After considering the different modules there different values for Accuracy, MAE and RMSE considered for checking the accuracy and efficiency of proposed system. The less values of MAE and RMSE and the high values of Accuracy shows the efficient working of the system proposed with the help of Artificial Neural Network.

<table>
<thead>
<tr>
<th>No. of particles</th>
<th>Starting module</th>
<th>Accuracy</th>
<th>MAE</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>30</td>
<td>97.71</td>
<td>12.57</td>
<td>15.97</td>
</tr>
<tr>
<td>20</td>
<td>60</td>
<td>97.95</td>
<td>12.47</td>
<td>15.98</td>
</tr>
<tr>
<td>30</td>
<td>70</td>
<td>98.84</td>
<td>12.65</td>
<td>16.2</td>
</tr>
<tr>
<td>40</td>
<td>50</td>
<td>97.7</td>
<td>11.83</td>
<td>14.93</td>
</tr>
<tr>
<td>50</td>
<td>50</td>
<td>97.6</td>
<td>12.75</td>
<td>16.73</td>
</tr>
</tbody>
</table>
7. Conclusion

From the literature review, it has been observed that, after deciding the architectural representation during the design phase while we are following some Model of software engineering, the choice of selecting a module from the architecture for coding and development is also very much important and we have to decide the sequence of selection of modules to go with the development. Any mistake during this selection process may lead to software crisis as if, after the implementation of some modules it is found that the section process was not good and we are not able to proceed further. So we have to start from the beginning. This may affect the overall cost of development, Time deadlines of the developers etc [14, 22]. So the selection sequence of development of modules is very important and somehow this aspect of software systems is not considered in the software implementation models. To perform the perfect selection of sequence of modules for development, we proposed a new technique with the help of Neural Networks, which helps to generate this sequence. The selection process depends on the properties of modules specified during design phase of software development cycle. These properties are coupling between...
the modules, Lack of Cohesion, etc. These properties will help to decide the weights of Neural Network and then with the help of Neural Network generation of sequence of execution of modules is completed. For the performance evaluation of the proposed system, I will use KC1 data set of NASA. This data set consist 22 different properties related to 2000 different modules. SNR, RMSE, CPU time and Accuracy are the parameters under consideration.

References


[20] Naeem Seliya, Member, IEEE, and Taghi M. Khoshgoftaar, Member, IEEE software Quality Analysis of Unlabeled Program Modules With Semisupervised Clustering


Conference on Quality Software,(APAQS 2000), Hong Kong, pp. 69-78.


Authors Profile

Ms. Manjot Kalsi received Msc (computer Science) degree from Khalsa College Amritsar and MCA from Kuruksherta University. She received Bsc. (Computer Science) degree from Baring Union Christian College Batala. She started her career as Assistant Professor at Baring Union Christian College Batala in department of Computer Science. She is pursuing Mtech (Computer Science & Engineering) from Lovely Professional University. She is presenting this paper as a research scholar.

Mr. Janpreet Singh received his B.Tech degree in IT from MIMIT, Malout (Punjab), INDIA, in 2002, and M.Tech degree in CSE from Guru Nanak Dev Engineering College, Ludhiana (Punjab), INDIA, in 2008. He is working as an Assistant Professor in Department of CSE at Lovely Professional University (Punjab), INDIA. His research interests include Databases and Software Engineering.