The Journal of Systems and Software 000 (2016) 1–21

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Hybrid functional link artificial neural network approach for predicting maintainability of object-oriented software☆

Lov Kumar*, Santanu Ku. Rath

Department of Computer Science & Engineering, National Institute of Technology, Rourkela, India

ARTICLE INFO

Article history:
Received 20 January 2015
Revised 12 August 2015
Accepted 2 January 2016
Available online xxx

Keywords:
Artificial neural network
CK metrics suite
Maintainability

ABSTRACT

In present day, software development methodology is mostly based on object-oriented paradigm. With the increase in the number of these software system, their effective maintenance aspects becomes a crucial factor. Most of the maintainability prediction models in literature are based on techniques such as regression analysis and simple neural network. In this paper, three artificial intelligence techniques (AI) such as hybrid approach of functional link artificial neural network (FLANN) with genetic algorithm (GA), particle swarm optimization (PSO) and clonal selection algorithm (CSA), i.e., FLANN-Genetic (FGA and AFGA), FLANN-PSO (FPSO and MFPSO), FLANN-CSA (FCSA) are applied to design a model for predicting maintainability. These three AI techniques are applied to predict maintainability on two case studies such as Quality Evaluation System (QUES) and User Interface System (UIMS). This paper also focuses on the effectiveness of feature reduction techniques such as rough set analysis (RSA) and principal component analysis (PCA) when they are applied for predicting maintainability. The results show that feature reduction techniques are very effective in obtaining better results while using FLANN-Genetic.

1. Introduction

Software developers often give more emphasis on object-oriented development methodologies, because of its inherent advantages over traditional development approaches. With the increasing complexity and size of object-oriented software, one major objective of software engineering approach is to have control over costs, schedules, and the quality of the software products. Several software quality attributes identified by ISO/IEC 9126 (Jung et al., 2004) can be listed as efficiency, functionality, maintainability, portability, reliability, and usability. In recent years, out of these several parameters, maintainability plays a high priority role for achieving considerable success in any software system. The ISO/IEC 9126 standard defines maintainability as the capability of the software product to be modified, including adaptation or improvements, corrections of the software to changes in environment and in requirements and functional specifications (Jung et al., 2004). In this paper, maintainability is considered as the number of source of lines changed per class. A line change can be an 'addition' or 'deletion' of lines of code in a class (Li and Henry, 1993).

As far as maintainability prediction is concerned, a number of unsolved problems and errors still are identified by researchers in the existing algorithms. Hence to obtain better results, it is essential to consider historical data of previous projects in a more empirical manner. Need of estimation of cost, effort, and various quality parameters has been always important for determining the feasibility of a project in terms of cost benefit analysis. In present day scenario, most of the project planning activities depend upon right estimates made at the early phase of software development life cycle (SDLC). Since class is the basic logical unit of an object-oriented system, emphasis needs to be given in assessing the quality of a class. Object-oriented software metrics can be of best help in such a scenario. So object-oriented software metrics can be considered in estimating certain properties of a class such as fault, effort, maintainability, re-work, productivity, etc.

Software system developed using object-oriented development methodologies support various features such as cohesion, coupling, inheritance, memory allocation etc. Some of the software metrics available in literature for measuring features of object-oriented system are as follows: Abreu MOOD metrics suite (Abreu and Carapuca, 1994; Melo and Abreu, 1996) Bansiya and Davis (QMOOD metrics suite) (Bansiya and Davis, 2002; Briand et al., 2000; Halstead, 1977; Henderson-Sellers, 1996; Lake and Cook, 1994; Lee et al., 1995; Li and Henry, 1993; Lorenz and Kidd, 1994; McCabe, 1976; Tegarden et al., 1995; Chidamber and Kemerer, 1994) suite etc. For study on estimation of maintainability parameter of a class,

☆ Fully documented templates are available in the elsarticle package on CTAN.

* Corresponding author. Tel.: +91 8260612874.

E-mail addresses: lovkumar505@gmail.com (L. Kumar), skrath@nitrkl.ac.in (S.Ku. Rath).

http://dx.doi.org/10.1016/j.jss.2016.01.003
0164-1212/© 2016 Elsevier Inc. All rights reserved.

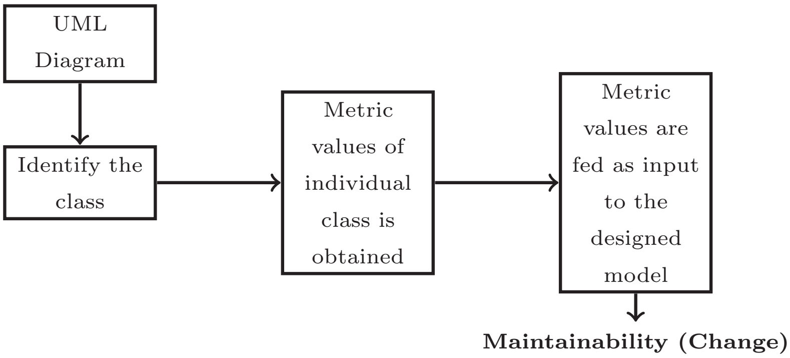
UML Diagram
Identify the class
Metric values of individual class is obtained
Metric values are fed as input to the designed model
Maintainability (Change)

Fig. 1. Flow chart For Maintainability prediction.

only those metrics are selected which have strong relationships with maintainability of software. CK metrics suite (Weighted Methods per Class (WMC), Depth of Inheritance (DIT), Number of Children (NOC), Lack of Cohesion in Methods (LCOM), and Response For a Class (RFC)), Li and Henry metrics (Message Passing Coupling (MPC), Data Abstraction Coupling (DAC), and Number of Local Methods (NOM)), and Size metrics (traditional line of code (SIZE1), and total number of attributes and methods of a class (SIZE2)) are selected for predicting maintainability of software system, as these three categories mostly emphasize on quality of a class.

Predicting maintainability of object-oriented software using software metrics has definite advantage because it helps to reduce future maintenance efforts and thereby improves the design aspect by providing information for more effective planning of different resources (Zhou and Leung, 2007). In order to predict the maintainability of a class, several traditional methods have been considered by authors as found in literature, but less importance has been given on application of machine learning techniques for this purpose. The AI techniques, a subset of machine learning methods have the ability of computer, software and firmware to measure the properties of a class, that human beings recognize as intelligent behavior. These methods are able to approximate the non-linear function with more precision. Hence they can be applied for predicting the quality parameter as maintainability in order to achieve better accuracy. In this paper, technique such as hybrid approach of functional link artificial neural network (FLANN) with genetic algorithm (FLANN-genetic or FGA), hybrid approach of functional link artificial neural network and particle swarm optimization (FLANN-PSO or FPSO), and hybrid approach of functional link artificial neural network and clonal selection algorithm (FLANN-CSA or FCSA) have been applied for maintainability prediction on two commercial software products such as Quality Evaluation System (QUES), and User Interface System (UIMS). These case studies are also considered by other authors in literature (Riaz et al., 2009). To train these models, object-oriented software metrics are fed as input data. This paper also focuses on the effectiveness of feature reduction techniques such as rough set analysis (RSA) and principal component analysis (PCA), which are used for reducing the complexity of estimation model by reducing number of input.

The generic steps to be followed to predict maintainability of any object-oriented software are shown in Fig. 1. Initially the classes of the respective software are identified from the class diagram and next, the different metric values of a class are extracted using different tools available in literature. Further, the metric values are considered as input to the designed model in order to predict maintainability of all individual classes in a software.

The remainder of the paper is organized as follows: Section 2 shows the related work in the field of software maintainability estimation, and object-oriented metrics. Section 3 highlights on research background related to this study. Section 4 briefs about the methodologies used to estimate the maintainability. Section 5 presents the performance parameters used for evaluating the models. Section 6 highlights on the results for maintainability prediction, achieved by applying techniques such as FGA, FPSO, and FCSA. Section 7 gives a note (comparison) on the performance of the designed models based on the performance parameters. In Section 8, various threats to validate have been discussed and Section 9 concludes the paper with comments on the models developed.

2. Related work

Various authors on design of prediction models which serve the purpose of computing the prediction rate in terms of accuracy for the parameter such as fault, effort, re-work, and maintainability have used software metrics as the basic input data, as shown in Table 1. In this paper, emphasis has also been given on this aspect of use of software metrics for maintainability prediction.

Table 1 shows the summary of literature review done on maintainability, where it describes the applicability of numerous software metrics used by various researchers and practitioners in designing their respective prediction models. From Table 1, it can be interpreted that a good number of authors have considered the use of statistical methods such as regression based analysis and their forms in predicting the quality parameter as maintainability. But keen observation reveals that work can be extended on use of neural network models for designing their respective prediction models.

Artificial neural network (ANN) models over the years have seen an explosion of interest, and their applicability across a wide range of problem domains. Indeed, ANN models are very often considered to solve problems related to prediction and classification. These models act as efficient predictors of dependent, and independent variables due to its characteristics to model complex functions. In this paper, software metrics are considered for predicting maintainability by applying three AI techniques.

3. Research background

The following subsections highlight on the data set used for computing maintainability. Data are normalized to obtain better accuracy and then dependent and independent variables are chosen for maintainability estimation.

3.1. Maintainability

In recent years, out of several quality parameters i.e., efficiency, functionality, maintainability, portability, reliability, and usability, maintainability have high priority in achieving considerable success for any software system. Table 2 shows the summary of

Please cite this article as: L. Kumar, S.Ku. Rath, Hybrid functional link artificial neural network approach for predicting maintainability of object-oriented software, The Journal of Systems and Software (2016), http://dx.doi.org/10.1016/j.jss.2016.01.003