

Chapter 24

Software Cost Estimation using Soft Computing Approaches

K. Vinaykumar

Institute for Development and Research in Banking Technology (IDRBT), India

V. Ravi

Institute for Development and Research in Banking Technology (IDRBT), India

Mahil Carr

Institute for Development and Research in Banking Technology (IDRBT), India

ABSTRACT

Software development has become an essential investment for many organizations. Software engineering practitioners have become more and more concerned about accurately predicting the cost of software products to be developed. Accurate estimates are desired but no model has proved to be successful at effectively and consistently predicting software development cost. This chapter investigates the use of the soft computing approaches in predicting the software development effort. Various statistical and intelligent techniques are employed to estimate software development effort. Further, based on the above-mentioned techniques, ensemble models are developed to forecast software development effort. Two types of ensemble models viz., linear (average) and nonlinear are designed and tested on COCOMO'81 data-set. Based on the experiments performed on the COCOMO'81 data, it was observed that the nonlinear ensemble using radial basis function network as arbitrator outperformed all the other ensembles and also the constituent statistical and intelligent techniques. The authors conclude that using soft computing models they can accurately estimate software development effort.

INTRODUCTION

Software development has become an important activity for many modern organizations (Pressman, 1997). In fact the quality, cost, and timeliness of developed software are often crucial determinants of an organization's success. There are significant financial and strategic implications for development projects in terms of activity scheduling and cost estimation. Software cost estimation is one of the most

DOI: 10.4018/978-1-60566-766-9.ch024

critical tasks in managing software projects. Development costs tend to increase with project complexity and hence accurate cost estimates are highly desired during the early stages of development (Wittig and Finnie, 1997). A major problem of the software cost estimation is first obtaining an accurate size estimate of the software to be developed (Kitchenham et al., 2003). An important objective of the software engineering community is to develop useful models that constructively explain the software development life cycle and accurately estimate the cost of software development.

In order to effectively develop software in an increasingly competitive and complex environment many organizations use software metrics as a part of their project management process. The field concerned with managing software development projects using empirical models is referred to as software project management (Fenton and Pleeger, 1997). Software metrics are aspects of software development (either of the software product itself, or of the development process producing that product) that can be measured. These measurements can be used as variables in models for predicting or estimating some aspects of the development process or product that are of interest.

Estimating development effort and schedule, can include activities such as assessing and predicting system quality, measuring system performance, estimating user satisfaction and in fact any modeling task involving measurable attributes of interest within the software development sphere (Gray, 1999). However, the most researched area has been effort estimation as it carries the greatest promise of benefit for project management. Such models are generally developed using a set of measures that describe the software development process, product and resources-such as developer experience, system size, complexity, and the characteristics of the development environment respectively. The output of the model is usually some measure of effort in terms of person hours (months or years).

There are many models and tools used in software cost estimation that provide invaluable information regarding efforts and expenditure to the management to bid for a project (Kitchenham et al., 2003). The most commonly used methods for predicting software development effort have been based on linear-least-squares regression such as COCOMO (Fenton and Pleeger, 1997; Pressman, 1997). As such, the models have been extremely susceptible to local variations in data points (Miyazaki et al., 1994). Additionally, the models have failed to deal with implicit nonlinearities and interactions between the characteristics of the project and effort (Gray, 1999).

In recent years, a number of alternative modeling techniques have been proposed. Existing data sets have their performance examined with some success including those in Gray and MacDonell (1997). Alternative models include artificial neural networks, analogy based reasoning, regression trees and rule induction models. Gray and MacDonnell (1997) applied fuzzy logic to software metric models for development effort estimation. They outlined the use of fuzzy logic for defining software metrics as linguistic variables and for modeling processes. They made comparison of results obtained from an elementary fuzzy inference system with other techniques such as linear regression and neural network techniques and found that it outperformed them well. Gray (1999) presented several different predictive model-building techniques such as robust statistical procedures, various forms of neural network models, fuzzy logic, case-based reasoning and regression trees. He also described a simulation-based study on the performance of these empirical modeling techniques using size and effort software metric dataset and observed that M-estimation regression outperformed all other parametric and non-parametric techniques. Xu and Khoshgoftaar (2004) presented an innovative fuzzy identification software cost estimation modeling technique, which is an advanced fuzzy logic technique that integrates fuzzy clustering, space projection, fuzzy inference and defuzzification. Based upon their case study on the COCOMO'81 database it was observed that the fuzzy identification model provided significantly better cost estimations

18 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/software-cost-estimation-using-soft/37001

Related Content

A Comparative Analysis on Economic Load Dispatch Problem Using Soft Computing Techniques
O.V. Singhand M. Singh (2020). *International Journal of Software Science and Computational Intelligence* (pp. 50-73).

www.irma-international.org/article/a-comparative-analysis-on-economic-load-dispatch-problem-using-soft-computing-techniques/252215

An Action Guided Constraint Satisfaction Technique for Planning Problem

Xiao Jiang, Pingyuan Cui, Rui Xu, Ai Gaoand Shengying Zhu (2016). *International Journal of Software Science and Computational Intelligence* (pp. 39-53).

www.irma-international.org/article/an-action-guided-constraint-satisfaction-technique-for-planning-problem/172126

Machine Learning Applications in Mega-Text Processing

Marina Sokolovaand Stan Szpakowicz (2010). *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques* (pp. 325-347).

www.irma-international.org/chapter/machine-learning-applications-mega-text/36992

Simulating Timing Behaviors for Cyber-Physical Systems Using Modelica

Hao Zhou, Mengyao Zhao, Linbo Wuand Xiaohong Chen (2019). *International Journal of Software Science and Computational Intelligence* (pp. 44-67).

www.irma-international.org/article/simulating-timing-behaviors-for-cyber-physical-systems-using-modelica/236151

Optimization of the Wake-Up Scheduling Using a Hybrid of Memetic and Tabu Search Algorithms for 3D-Wireless Sensor Networks

Vrajesh Kumar Chawraand Govind P. Gupta (2022). *International Journal of Software Science and Computational Intelligence* (pp. 1-18).

www.irma-international.org/article/optimization-of-the-wake-up-scheduling-using-a-hybrid-of-memetic-and-tabu-search-algorithms-for-3d-wireless-sensor-networks/300359