Fuzzy logic technique for estimating software cost using Cocomo

Nishi
nishisinghal54@gmail.com
Bhagat Phool Singh Mahila Vishwavidyalaya,
Sonepat, Haryana

Vikas Malik
vikassmalik@gmail.com
Bhagat Phool Singh Mahila Vishwavidyalaya,
Sonepat, Haryana

ABSTRACT

Software cost estimation is the most challenging and important activities in software development. Developers want a simple and accurate method of efforts estimation. Estimation of cost before starting of work is a prediction and prediction always not accurate. The exact Software Price Estimation (SPE) is very difficult in the early stages of software development because many input parameters that are active in software exploit are uncertain in the early stages. Accurate cost estimation helps us complete the project within time and budget. For this work, we have knowledge of all available techniques and tool of software cost estimation. The growing application of software and resource constraints in software task development needs a more accurate estimate of the cost and effort because of the importance in program planning and resource management admitting the number of programmings and software plan using tools and modern methods of modeling. The fuzzy set and fuzzy logic can produce accurate software attributes.

Keywords*: Software Cost Estimation, Software cost estimation models, Software cost estimation technique, Fuzzy Logic for Software Cost Estimation.*

1. INTRODUCTION

Software cost estimation acts an important role in software engineering. Software engineering research deals with various aspects of software development. Software cost estimation remains one of the most critical areas of research due to the financial aspects involved. A software price estimate process is the set of proficiencies and procedures that organizations use to arrive at an estimate. Software cost estimation is applied in various government and non-government organizations, defense organizations, aeronautics, etc. Accurate cost estimates activity is critical to developers and customers [1]. Accurate cost estimation is important for following reasons

- It can be used to classify and prioritize development projects with respect to complete business plan.
- It can help to find out what resources to commit to the project and how well these resources will be used.
- It can help to assess the impact of changes and supporting for poor planning. Projects can be easier to manage and control when resources are matched to real needs.
- Customers expect accurate development costs to be in line with estimated costs.

This paper offers an introductory view of young Practitioners to providing detailed about software cost estimation methods and models. There are various techniques and models, we use for the software cost estimation, but which method is suitable for cost estimation it's very difficult to decide. To solve this type trouble it is very necessary to know about the software cost estimation methods and models [2]. Effort estimation is an essential part of software development, for the favourable result of software. As the software produces in size and complexes the software effort estimation task is getting complex there for to the lot with the complexity since last few years, many researchers all terminated the world efforts to develop new modelling proficiencies which could deal with the modifying complexity and increased size of software.

2. LITERATURE SURVEY

For the estimation of Software cost, several researches have been suggested by researchers. This study examines the fundamentals of software cost and estimation. Different techniques of cost estimation should be employed when estimating costs. Estimates are

made to find out the cost, to the developer, of producing a software system. Following are the literatures used for evaluation of the state-of-art work on the estimation of software Cost [3]. Increased use of information system has led to bigger amount of developing expenses which insists on software. Using SVR (Support Vector Regression), a sort of machine learning technique (Lee & Kwon 2009), the software cost is estimated approximately. This finds the best set of parameters applying immune algorithm. The Software cost estimation was performed by SVR based on immune algorithm while changing populations, memory cells, and the number of allele (two or more forms of a gene) [4]. Moreover, they have examined and compared the effects of software cost estimation by means of the optimal set of parameters found by immune algorithm and showed that software cost estimation using SVR based on immune algorithm was superior to linear regression model and other machine learning methods like analogy, artificial neural network, and genetic programming. Estimation of cost for any project was the most important and distinctive matter for the project managers because of lack of data during the early stages of software development and intrinsic uncertainties which are related with it [5]. In spite of huge number of suggested algorithmic models, the science of cost estimation yet lacks accuracy because of their incapability to handle with these uncertainties. To triumph over this circumstance, concentration has been moved towards soft computing techniques. Kashyap & Misra (2013) have employed Quality Function Deployment (QFD) technique to promote a good understanding among customer requirements and design specification. This distinctive mixture will assist the project managers to competently plan the overall software development life cycle of the software product. Hence, it presents a better cost estimate. The combination of QFD technique and PSO methods are incorporated to increase better accuracy in software cost estimation.

3. SOFTWARE COST ESTIMATION

SCE for software projects includes coordination between all developmental activities, design, production monitoring, maintenance, etc. Accurate estimation of software project causes that internal and external processes and employee activities, efforts and costs to be coordinated. So, before pattern and implementation of software projects offering the model for them is essential and can be the most difficult tasks in software task development. In the process of software task's output to reduce cost and schedule and probabilistic risk appraisal must be taken to avoid project failure [6]. The importance of SCE is more apparent when we know each evaluation in the estimation of price contains the positive and negative results and its balancing at any point of time is single of the most complex management issues. Evaluation of the ideas performance has always been the important and valuable category in software development projects. So the more accurate the criterion of evaluation of software projects, the better estimate would be achieved.

Software Cost Estimation Models

(A). SEL Model: The Software Engineering Laboratory (SEL) of the University of Maryland has established a model i.e. SEL Model for estimation

(B). Walston-Felix Model: WFM created by Walston and Felix at IBM which provides a relationship between source code and effort E. This model establishes various aspects of the software growth environment such as user participation, customer-oriented changes, memory constraints etc.

(C).COCOMO Model: Canonic COCOMO enumerates software development exploits as a subroutine of program size. Program size is obvious in estimated thousands of SLOC. COCOMO uses three divisions of software tasks

(D) Intermediate COCOMO Model: According to this model effort play an important role of program size (PS) and a set of "cost drivers" that include subjective assessment of the product, hardware, personnel and project attributes. This extension considers a set of four "cost drivers" which are Product attributes, Hardware attributes, Personnel attributes and Project attributes

(E). Detailed COCOMO Model: This model having all the crucial characteristics of previous COCOMO model which is an intermediate version with an evaluation of the cost driver's impact on each step of the software engineering process. In detailed COCOMO, the effort is calculated as part of program size and a set of cost drivers given according to each phase of software life cycle [7]. The five phases of detailed COCOMO are:

1) Plan and requirement.
2) System design.
3) Detailed design.
4) Module code and test.
5) Integration and test.

Therefore we can say that COCOMO is playing a very important role to estimate the cost of software accurately because it is a very tedious task and various risk factors are involved. If decision would be wrong then huge loss occurred to the organization. To derive it various parameters must be kept in mind for example scale factors and efforts multipliers.

4. ESTIMATION TECHNIQUE

There are many methods and proficiency for software cost estimation which are separated into two groups which are known as Non-Algorithmic and Algorithmic. In the Algorithmic method, we apply a formula for calculating the cost estimation. The rule is created by combining related cost factors in the various models. Non-algorithmic techniques unable to use a formula to evaluate the software cost estimation. Both groups are useful for performing the accurate estimation. If the requirements are known better, their performance will be better.

Non-Algorithmic Models: These models first compare the project under consideration with the previously done projects by the organization and analyses the information from the most similar projects to make the cost estimates. Basically, these methods make use of the past experiences.

(A). Analogy-Based Estimation: Estimation by Analogy process to differentiate the project at hand with preceding executed projects which are similar to the proposed project and extrapolate the information from those projects to estimate its cost. It has been seen that experts often apply analogy reasoning while making cost estimates. This method is simple and flexible. Since it relies on the historical data, it can help in dealing with poorly understood areas. It can be applied at the initial phases of SDLC and can later be improved upon as more information about the proposed project becomes available [8]. The process for analogy based estimation goes as follows:

- Identify the requirements of the current project. Select the most similar projects completed previously which have their information in the database.
- Extrapolate this data to estimate the cost of the proposed project.

(B). Expert Judgment Method: Expert judgment proficiencies take consulting with cost estimation experts or a group of the approximate experts to use their experience and reading of the proposed project to generate in at an estimate of the price. It is the almost usable methods for the cost approximation. Most companies exploited expert judgment method for getting the cost of the product. This method applies next estimating steps   1. Project leader resents each expert with a specification and an appraisal form.
2. The experts fill out forms anonymously
3. Project leader calls a group meeting in which the experts discuss cost estimation issues with the project leader and each other.
4. Project leader prepares and distributes a summary of the cost estimation on an iteration form.
5. Again experts fill out forms, anonymously.
6. Steps D and Step E are iterated for as many rounds as appropriate.

(C). Price-to-win: The price is estimated to be the best cost to win the project. The cost estimate is founded on the customer's budget rather of the software functionality. For example, if a sensible estimate for a project costs 100 PM only the customer can only effort 60 PM. It is common that the estimator is asked to change the estimation to fit 60 PM effort in order to win the project. This is again not a good practice since it is very likely to cause a bad delay of delivery or force the estimation team to work overtime.

(D). Parkinson's Law: Using Parkinson's Law "Work expands to fill the available volume", the cost is determined by the usable resources rather than based on an objective assessment. If the software has to be presented at 20 months and 4 people are available, the effort is estimated to be 80 PM Although it sometimes gives a good estimation, this method is not urged as it may provide very unrealistic estimates. Parkinson's Law does not promote good software engineering practice.

Algorithmic Models: Cost estimation model get by using the algorithmic method, this displays the importance of the algorithm based model in the files of software cost estimation. The algorithmic method involves the use of equations to execute software Estimates.

(A). PUTNAM MODEL: The Putnam model is very tender to the development time can greatly gain the person-months needed for evolution. One substantial problem with the PUTNAM model is that it is founded on experience, or being able to estimate exactly, the size (in lines of code) of the software to be developed. There is much greater doubt on the software size. It may result in the inaccuracy of cost estimation [9].

(B). COCOMO Models: One very widely used algorithmic software price model is the Constructive Cost Model (COCOMO). The basic COCOMO model has a very elementary form. MAN-MONTHS = K1* (Thousands of Delivered Source Instructions) K2 Where K1 and K2 are two parameters dependent on the diligence and evolution environment. Estimates from the canonical COCOMO model can be made more accurate by continuing into account other factors interesting the required characteristics of the software to be produced, the qualification and experience of the development team, and the software development environment.

(C). Seer-Sem Models: This model has been proposed in 1980 by Galorath Inc. Most parameters in seer-them are commercial and, business projects usually use seer- SEM as their main cost estimation method. The size of the software is the most important feature in sear- same method and parameters, namely, user is defined as efficient size. SEM is computed by determining the five indicators: new size, existing size, re imp re-plan and retest.

5. FUZZY LOGIC FOR COST ESTIMATION

Fuzzy Logic is a superset of a Boolean logic and that has been extended to take care of the partly truth values. It is a mathematical technique for dealing with imprecise data and problems that have many solutions rather than one. Fuzzy logic works with scopes of values, solving problems in a mode that more resembles human logic. Fuzzy logic is a coherent system which is a reference to multi-valued logic. The fuzzy system changes these rules to their mathematical equivalents. Fuzzy logic can hold problems with imprecise, undefined and incomplete data. In Fuzzy Logic, a proposal may be true or false or have an average truth value, such as may be true. Fuzzy systems try to emulate the cognitive operation of the brain with a rule base. The basic concept is inspired by the human process, where the decision criteria are not clear-cut simply blurred and is difficult to find objective to make the

decision more precise and clear [10]. The fuzzy logic theory was presented in 1965 because of the uncertainty in data and information and imprecision in the existence of vagueness. Fuzzy Logic is not a random or unlikely method and in fact, this method itself introduces a special system to deal with the ambiguous and non-deterministic situations. The essential characteristic of the fuzzy theory is displaying uncertain data and also can be the operation and application of mathematical programming. Fuzzy logic is used to solve the complex problems, which is based on fuzzy set theory introduced by Prof. Zadhe in using fuzzy logic linguistic constructs can be used, i.e. fuzzy logic provides the inference structure which enables human reasoning capabilities. The fuzzy logic systems can be of three categories as pure Fuzzy, Takagi fuzzy and Surgeon Fuzzy, and fuzzy system with falsifier and defuzzifier out of the three the fuzzy system with falsifier and defuzzifier is the most popularly used which is proposed my Mamdani, it is the combination of four main components as Fuzzifier, Fuzzy Rule base, Fuzzy inference engine, and Defuzzification. Fuzzifier converts a crisp input to a fuzzy set, rule base uses the fuzzy if-then rules, inference engine falsifies the linguistics values and Defuzzification converts the fuzzy output into the crisp result. Membership functions are used in Fuzzification and Defuzzification steps of a fuzzy logic system for the purpose of mapping fuzzy logic linguistic terms and non-fuzzy inputs or vice a versa, membership functions are used for quantification of linguistic terms [11]. In many engineering sciences lots of quality and quantity factors such as quality, cost, flexibility, scalability, and operation must be considered in decision-making. To do so would be to find out the factors and weights of fuzzy functions use and fuzzy numbers can be explicit in them. So Fuzzy Logic tries to hold a convenient option for issues with estimates and decision making in environments with ambiguous and undefined criteria. In recent years, Fuzzy Logic has many diligences in Software Cost Estimation, due to the flexibility and high precision in the estimates [12]. In this section, we review FL models and its applications and also the modeling output of proposed models that have been done by the researchers on the project software dataset. The proposed method is combined with COCOMO II model.

6. SOFTWARE USED

Software: MATLAB Version R2015a: It is powerful software that provides an environment for numerical computation as well as a graphical display of outputs. In Matlab, the data input is in the ASCII format as well as binary format. It is a superlative language for technical computing integrates computation, visualization, and programming in a simple way where problems and solutions are expressed in familiar mathematical notation.

- Acquisition, Data Exploration, Analysing &Visualization
- Engineering complex drawing and scientific graphics
- Analysing of algorithmic designing
- Mathematical and Computational functions
- Modeling and simulating problems prototyping
- GUI (graphical user interface) building environment.

Using MATLAB, you can solve technical computing problems very easily and time-saving as compared to traditional programming languages, such as C, C++, and FORTRAN.

7. CONCLUSION

A single important event in software project management is exact and authentic estimation of software time, price, and workforce, especially in the former phase of software evolution. Software attributes usually have properties of doubt and vagueness when they are measured by human judgment. A software cost estimation model contains fuzzy logic can overcome the uncertainty and vagueness of software attributes. In this paper, was investigated using the Fuzzy Logic and the capacities of COCOMO model. The use of Fuzzy Logic as a new advance in analyzing software engineering events such as software price estimation model has amended. One of the issues that helps the correct usage of software and preclude the software projects from failure is the accurate estimate of cost. The fuzzy logic used for originating the model is sufficiently general and can be applied to other models of software cost estimation models based on function point methods and to other areas of quantitative software engineering. To produce a meaningful and reliable estimate, we must improve our understanding of software project attributes and their causal relationships, develop effective ways of measuring software complexity and the cost estimation process needs to be thoroughly ordered and carefully followed. The main aim of this paper to provide a detailed survey or introduction of available cost estimation models and technique

8. REFERENCES

[1] Ravishankar.S, P. Latha, "Software Cost Estimation using Fuzzy Logic", International Conference on Recent Trends in Communication and Controls (ICON3C 2012) in International Journal of Computer Applications® (IJCA).
[2] Anish Mittal, Kamal Parkash, Harish Mittal." Software Cost Estimation Using Fuzzy Logic", Page 1, November 2010, Volume 35, Number 1.
[3] Isa Maleki, Laya Ebrahim, Saman Jodati, Iraj Ramesh," ANALYSIS OF SOFTWARE COST ESTIMATION USING FUZZY LOGIC", International Journal in Foundations of Computer Science & Technology (IJFCST), Vol.4, No.3, May 2014.
[4] Ravneet Preet Singh Bedi and Amardeep Singh," Software Cost Estimation using Fuzzy Logic Technique", Indian Journal of Science and Technology, Vol 10(3), /ijst/2017/v10i3/109997, January 2017.
[5] Rekha Tripathi, Dr. P. K. Rai Study Centre for Computer Application," Comparative Study of Software Cost Estimation Techniques", International Journal of Advanced Research in Computer Science and software engineering, Volume 6, Issue 1, January 2016.
[6] Rekha Tripathi, Dr. P. K. Rai Study Centre for Computer Application," Comparative Study of Software Cost Estimation Techniques", International Journal of Advanced Research in Computer Science and software engineering, Volume 6, Issue 1, January 2016.

[7] Narendra Sharm, Aman Bajpai, Mr. Ratnesh Litoriya," A comparison of software cost estimation methods: A Survey", The International Journal of Computer Science & Applications (TIJCSA), Volume 1, No. 3, May 2012 ISSN – 2278-1080.

[8] Tannu, Yogesh Kumar," Comparative Analysis of Different Software Cost Estimation Methods", International Journal of Computer Science and Mobile Computing, IJCSMC, Vol. 3, Issue. 6, June 2014, pg.547 – 557.

[9] Sanjali Gupta, Sarthak Tiwari, Himanshu Singh, Ayush Shukla, Himanshi Raghuvanshi, " A Comparison between Various Software Cost Estimation Models", International Journal of Emerging Trends in Science and Technology,IJETST-Vol.||03||Issue||11||Pages4771-4776||November||ISSN 2348-9480.

[10] T.N.Sharma, Anil Bhardwaj, Anita Sharma," A Comparative study of COCOMO II and Putnam models of Software Cost Estimation", International Journal of Scientific & Engineering Research, Volume 2, Issue 11, November-2011 1 ISSN 2229-5518 IJSER © 2011 http://www .ijser.org

[11] Josephin. M. Baby Anusha M.E, Rajeswari Mukesh M.E, Ph.D.," A Study On Software Cost Estimation", National Conference on Architecture, Software systems and Green computing-2013(NCASG2013).

[12] Mr. Azade Sanjay Y, Dr. Lomte S. S., Mr. Aqueel Ahmed A. Jalil," Soft Computing Techniques for Software Effort Estimation Comparative Study Using Cocomo Ii Model", IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, PP 45-50 www.iosrjournals.org.