ISSN 2348 - 9928
IJAICT Volume 7, Issue 10, October 2020 Doi:10.46532/ijaict-2020033 Published on 05 (10) 2020

Insights on Cost Estimation Methods and its Uses in Software Project Design

[1] **Aravindhan K**
[1]Department of Computer Science and Engineering, SNS College of Engineering, Tamil Nadu, India.
[1]aravindhan03@gmail.com

ArticleInfo
International Journal of Advanced Information and Communication Technology
(https://www.ijaict.com/journals/ijaict/ijaict_home.html)
https://doi.org/10.46532/ijaict-2020033
Received 03 May 2020; Revised form 30 July 2020; Accepted 23 September 2020; Available online 05 October 2020.

Abstract- Cost estimation of software projects is risky task in project management field. It is a process of predicting the cost and effort required to develop a software applications. Several cost estimation models have been proposed over the last thirty to forty years. Many software companies track and analyse the current project by measuring the planed cost and estimate the accuracy. If the estimation is not proper then it leads to the failure of the project. One of the challenging tasks in project management is how to evaluate the different cost estimation and selecting the proper model for the current project. This paper summarizes the different cost estimation model and its techniques. It also provides the proper model selection for the different types of the projects.

Keywords - Software cost estimation; COCOMO; Project management; Algorithmic and Non-Algorithmic Methods.

1. Introduction

With the advancement in the field of software development, Software cost estimation has become more critical. Software Cost estimation [1] is the process of evaluating the effort required in developing particular software [2]. Cost estimation is required for many reasons, a few of which are noted bellow:

1. The first and foremost requirement of cost estimation is that we can give a rough idea of how much capital investment will be required to successfully conclude the project.
2. It gives an idea of resources required in various development phase. For example our project deals with customers having less technical knowledge, in that case, we need more testing hence the cost of this phase will be comparatively higher.
3. We will be able to deal with the changing requirements and this will help us to either fulfil the requirement in the given budget or give a new estimation to the client.
4. The management of the project becomes easier, since we know what the end goal is and we have fair idea how to achieve the end goal utilizing the resources on hand.
5. The chances of failures decrease drastically if we have good estimate of the project since we allocate the resources accordingly.

Hence it becomes utmost important to estimate the cost of project. This estimation should be as accurate as possible as over estimation and underestimation of the cost also has adverse effects. It is very hard to estimate the correct cost of project. Numerous estimation methods are available and are used depending on the type of software being developed. All these methods are derived from certain basic models which are modified and refined over a period of time in order to get more accurate and reliable estimations. The paper focuses on these basic methods of cost estimation.

2. Estimation methods

Certain methods are proposed to predict the cost of software. These estimation methods can be broadly categorized as follows

1. Non Algorithms Methods
2. Algorithmic methods

The advantage and disadvantages of utilizing each method are covered below.

Non Algorithmic Methods

Non algorithmic methods need to historical data to estimate the cost of the software. This historical data is obtained from the similar projects carried out in past. The estimation is done using this data and keeping in mind other external factors which might affect the cost of project.

Historical estimation methods

As the name suggests these methods use historical data to predict the cost of the on-going project. The estimate is done on the basis of the documentation available from the previous similar projects. The estimation is done using the resources utilized in completing that project and also the time taken to complete the project having the resource in hand. Thus this becomes the simplest model to estimate the cost of the project. The main advantage of this method is its reliability. If the documentation or the source of the data is to the mark then this method gives a very accurate estimate of the cost which will incur to finish the development. However inaccurate data might adversely affect the estimation. Hence a validated reliable data source has to be used for this method. Accuracy of the model also depends on which project's data is used for estimation. The selected data should be coming from the project which is highly analogous to the project whose cost is to be estimated. This helps giving much more reliable estimation. One major disadvantage in using this model is that the data for all the projects will not available. This might be due to improper documentation or changing requirements. The requirements of the project to be estimated can be completely new and historical data might not be available, in such case this model will not be able to give us a good estimation

Expert's Judgement

This model is used very frequently in industry. Industry veteran working in the same domain is concerned and estimation is made. This model purely depends on the experience of the expert of dealing with similar project in the past. The advantage of this model is that, the expert can clearly differentiate and point out the heterogeneity in the project on hand and the project dealt in past. Expert can also point out the areas in which the estimation might vary and also how to optimally allocate and utilize the resources. The problem faced here is that the experience is not quantifiable resulting in error of estimation. The expert can be biased too and because of this the correct estimation might not be provided. There is no standard set for selecting an expert too. However if the right expert is chosen then, this method will yield most accurate estimation. Therefore, this model is always used in complementary to other models [10].

3. Delphi method

Delphi method can be considered as the refinement of expert's judgment [5]. It deals with the standardization of the expert's judgement model. There will be form shared to carry out this process. The form will have the questions about the parameters on which estimation is to be made. It is important to note that Delphi method is an iterative process, the steps involved are as follows;

i. Experts are briefed about the product and the parameters which are to be estimated. These parameters can be time, money, resources, technologies etc.
ii. Each expert will produce an individual review.
iii. The result of the questionnaire will be generated in such a way that it will show individual review of the expert and also the group review.
iv. The results are then discussed with the experts again.
v. The experts give an independent review again after discussion.
vi. The steps 3rd to 5th are repeated until we get an estimate which agreed upon by all the experts.

The main advantage of this method is that it reduces the biasness that was pertaining in expert's judgement model. Experts are able to share their knowledge with each other. During the discussion period, experts can reason among one another this will increase the accuracy of estimation. All the aspects related to the project will be brought out and this will lead to lesser deviation from the estimated cost. However this process is time consuming. This is because it might take certain amount of time before all the experts agree to an estimated value of a certain parameter. The number of experts to be selected is also not defined and standardized. The estimation accuracy can be compromised due to the internal conflicts between the experts. The questionnaire which is to be prepared has to be clear and precise, so that the inputs derived from the questionnaire serve the desired purpose.

Bottom up approach

Bottom up approach of software cost estimation [3] deals with cost estimation of each component independently. For this model to give a reliable estimate it is necessary that our project has a clear higher level design. The exact number of components in the project must be clear. The main advantage of this model is that it makes the estimation very easy because of its simplicity. The disadvantage of this model is the collaboration between teams. The model also oversees the cost of integration and it takes lot of time because estimation process is run for each of the components. The model also is not viable for projects having a lot of components.

Price to Win Estimation

The main focus of this estimation is to meet the budget or price range given by the customer. Functionality of the software is not the first priority. The functionality is modified according to the price range. The advantage of this estimation method is that the customer is satisfied since the given budget is met. However the method will prove to be inefficient in case of changing requirements. The method also proves to be slow due to which the developer may suffer loss.

Top down Approach

Top down approach is opposite of bottom up approach. Here we assume or estimate the cost of entire project rather than, estimating the cost of individual components. The estimation can be carried out at earlier stages and its lot faster as we avoid the individual details of each component. The estimation can be done with least amount of detail using historical data or algorithmic approach. However the disadvantage of using this model is that it misses lower level system complication which might incur in the project. The estimation is fast using this model and this model is well suited for system having less number of components. The cost of integration and other system level implementations are also covered in this model unlike bottom up approach [11].

Artificial Neural Network based estimation

Artificial neural networks are trained using historical data and then the prediction of the software cost is carried out. The prediction done using ANN's [6] reduces the gap between the actual cost and estimated cost. Estimation provided using this method is reliable and it adds the power of reasoning in the estimation unlike that of estimation done using historical methods in which estimation is done only using historical data. Computational power required in this method is quite high. Large amounts of data are needed to train ANN's. There is no particular procedure specified to design the ANN.

Fuzzy Logic based estimation

Fuzzy logic based estimation [7] also known as soft computing estimation is proposed to overcome difficulty in using the existing methods. Software development process is not particular and the parameters involved might not quantifiable. However such parameters might be necessary for correct estimation. Therefore, fuzzy logic based estimation methods provide an advantage as these methods were proposed to deal with problem statements whose mathematical model was not easy to create. The main advantage of this method is that we can express non quantifiable parameters or fuzzy parameters using this method and also no special training is required to utilize this method. However modelling complex systems becomes difficult using this method because of involvement of many fuzzy parameters.

4. Algorithmic method

The algorithmic method uses mathematical equations for cost estimation. The equations are based on the data available from the past and have various sources of inputs. These methods are widely researched and various popular model like cocomo model, Putnam model and function based model are developed using this models. The major advantage of this model is the ease of carrying out the estimation. Various parameters involved with the estimation are converted as input to the equations and estimation is carried out. The changed estimate can be calculated easily as the model already use quantifiable input parameters. The disadvantage of using this model is the lack of the actual developmental inputs necessary of any estimates. The quantification of the parameters is also not an easy task.

COCOMO (Constructive Cost Model)

Initially proposed by Boehm in 1981 COCOMO model became one of the widely used models soon after its release. It was a popular estimation model till 1987 after which the initial model proposed by Boehm in 1981 also commonly known as COCOMO'81 started facing trouble. This was due to the introduction of new development methodologies which promoted reusability; rapid development and usage of object oriented concepts in development. Hence COCOMO II was proposed which provided a solution to the new upcoming technologies and hence helped the software developers to give more and more reliable estimation. COCOMO II [3] deals with the newly software development analogies using three sub models. The equations used in the COCOMO model and their implementation are available making it much easier to utilize this model for cost estimation. The model is widely used throughout the industry. It deals with parameters like code size which is measured in KLOC [4] and also the personnel required completing the model and hence it is much related to the original development process. COCOMO is a regression model which was developed using 63 models. One of the drawbacks of using this model is that it is difficult to get estimate of the lines of codes required at earlier stages.

Putnam model

The estimation is based on the distribution of man power. The idea of which is obtained using previously conducted projects. The equation used in this model is as follows

$S = E * \text{Effort}^{1/3} * t^{3/4}$

Where

S: Size of the software

E: External factors or environmental factors affecting the project

t: time required to deliver the fully working software

Effort: measured person year

Putnam also defined the effort using the following relation

$\text{Effort}= D * t^3$

D is the manpower build up factor. The tool used for estimation of cost and manpower measurement is SLIM based on Putnam's model. Advantage of using this method is that it focuses on two important parameters of software development which is time and size using which estimation is done on manpower and the effort required. However other aspects of software development are not considered.

Function point Based analysis

The estimation given by this method is based on functionality of the software. The model assumes the functionality based on logic, interfaces, input output parameters etc. The model utilizes the functional metric for cost estimation [8]. The prediction done on functional points of software is independent of language coded or tools used; therefore estimation can be given at earlier stages. The estimation provided is better than SLOC. The estimation is more time consuming since it requires manual work. Developers who lack experience cannot predict the cost of the software at earlier stages.

Software Estimation

The software estimation or cost estimation deals with several factors like the cost of completing the project, time taken, human resources required etc. It is done with the models which are summarized above. The estimation process starts with the selection of the estimation method. This selection depends on various factors which are contributing to the software development and the project in consideration. Going through the advantages and disadvantages of the models which are described above, it is safe to say that accurate prediction might not be possible using a single model. The algorithmic models though providing a higher amount of flexibility cannot be trusted individually as they lack the practical experience. Similarly the models like expert's judgement and Delphi methods provide necessary practical exposure but can be time consuming and biased. It is recommended to use the models complementing each other and thus striving towards more precise estimations. The selection of the model is only half the issues. Execution or implementation of the model is of equal concern. In any software projects the requirements keep on changing [9]. It is essential to estimate the cost after change in even minute requirement. The estimation rather should be a continuous process which is spread across all the development phases which will yield a higher throughput as the resources utilized till now and the resources which are to further utilized are kept in mind. The estimation should be carried out with all the models selected earlier. The parameters which are to be estimated considered should be correctly defined thus giving a clearer idea to the personnel carrying out the estimation in Table 1.

Table 1. Summary of Existing Techniques of cost estimation

Cost estimation models and their advantages and disadvantages		
Methods	**Strengths**	**Weakness**
Expert Judgement [5]	Good practical exposure and all the precaution points are covered	Can be biased
Historical Data	Estimation based on previous projects	The data might not be available
Top-Down	Faster to give implementation	Less precision compared to other method
Bottom-up	Estimation of each module and easy to implement	Certain higher level costs are ignored
Delphi	Unbiased opinion from various experts	Time consuming
Price to Win Estimation	Highly satisfied customer	Tends to be slow and developer may suffer loss
Artificial Neural Network based estimation [6]	Reliable estimate than estimates based only on historical data	Requires tremendous amount of computational power and huge dataset
Fuzzy Logic based estimation [7]	Easier to express parameters which are not clear	Difficult to model complex systems.
COCOMO	Simple to use	Lines of code estimation difficult at earlier stages
Putnam Model	Time and size estimation is obtained	All developmental stages and aspects not covered
Function Point Based analysis[8]	Reliable estimate than SLOC	Not suitable for developers lacking experience

5. Conclusion

The cost estimation of software is an important process. It is carried out using various models which gives a rough estimation of the cost which will be incurred. The pros and cons of every model are discussed in the paper and using these inferences we can safely state that it is not possible to accurately specify the cost using any single model. However it is important to note that these models provides the necessary framework needed for estimation and the quality of the estimation can be improved using combination of models. The quality of estimation can be further improved by keeping certain small pointers such checking the validity of the data used, concerning with more than expert, conducting the reviews multiple times at different phases etc. These basic heuristics can improve the precision of estimation.

References

[1]. Pressman, Roger S."Software Engineering: A Practitioner's Approach",6th Edition,McGraw- Hill, New York,USA,ISBN :13:9780073019338,2005.

[2]. E. Kocaguneli, T. Menzies, J. Keung, D. Cok and R. Madachy, "Active Learning and Effort Estimation: Finding the Essential Content of Software Effort Estimation Data," in IEEE Transactions on Software Engineering, vol. 39, no. 8, pp. 1040-1053, 2013.

[3]. G.K. Mislick, D. A. Nussbaum, "Cost Estimation: Methods and Tools", John Wiley & Sons, 2015.

[4]. E. Morozoff, "Using a Line of Code Metric to Understand Software Rework," in IEEE Software, vol. 27, no. 1, pp. 72-77, Jan.-Feb. 2010.

[5]. M.Jorgensena, "Review of studies on Expert Estimation of Software Development Effort", Journal of Systems and Software,70(1-2):37-60,2004.

[6]. Attarzadeh, I.Siew Hock Ow,"Proposing a New Software Cost Estimation Model Based on Artificial Neural Networks", IEEE International Conference on Computer Engineering and Technology (ICCET) Volume: 3, Page(s): V3-487 – V3-491,2010.

[7]. H. Anandakumar and K. Umamaheswari, "Supervised machine learning techniques in cognitive radio networks during cooperative spectrum handovers," Cluster Computing, vol. 20, no. 2, pp. 1505–1515, Mar. 2017.

[8]. H. Anandakumar and K. Umamaheswari, "A bio-inspired swarm intelligence technique for social aware cognitive radio handovers," Computers & Electrical Engineering, vol. 71, pp. 925–937, Oct. 2018. doi:10.1016/j.compeleceng.2017.09.016

[9]. R. Arulmurugan and H. Anandakumar, "Early Detection of Lung Cancer Using Wavelet Feature Descriptor and Feed Forward Back Propagation Neural Networks Classifier," Lecture Notes in Computational Vision and Biomechanics, pp. 103–110, 2018. doi:10.1007/978-3-319-71767-8_9

[10]. Haldorai, A. Ramu, and S. Murugan, "Social Aware Cognitive Radio Networks," Social Network Analytics for Contemporary Business Organizations, pp. 188–202. doi:10.4018/978-1-5225-5097-6.ch010

[11]. R. Arulmurugan and H. Anandakumar, "Region-based seed point cell segmentation and detection for biomedical image analysis," International Journal of Biomedical Engineering and Technology, vol. 27, no. 4, p. 273, 2018.