ISSN No. 0976-5697

Volume 8, No. 5, May-June 2017

International Journal of Advanced Research in Computer Science

REVIEW ARTICLE

Available Online at www.ijarcs.info

Software Effort Estimation in Early Stages of Software Development: A Review

Mudasir M Kirmani
Assistant Professor, Computer Science
Sher-e-Kashmir University of Agricultural Sciences & Technology of Kashmir
Srinagar, J&K, India

Abstract: Cost overruns can cause customers to cancel projects whereas the cost underestimates can force a project team to invest much of its time without financial compensation. The accurate prediction of software development costs is a critical issue to make sound management decisions and accurately determining how much effort and time a project requires for project managers, system analysts and developers. Without reasonably accurate cost estimation capability, project managers cannot determine how much time, manpower, cost and resources the project should take from beginning of a project till its completion.

During the past ten to fifteen years, the importance of software in general and web application in particular has dramatically increased in our daily lives. This trend is expected to accelerate in the coming years, as software's importance in our day to day activities has grown and the focus on its overall performance both technically and programmatically has also increased. A good cost estimate early in the project's life cycle helps the project manager to optimally implement the project within budget and scheduled time. This research work is an effort to review different methods of effort estimation and to highlight the advantages and gaps of a model proposed earlier.

Keywords: software, effort, estimation

1. INTRODUCTION

The concept of sharing information on a platform which will be stored at one location and visible to all started way back in 1989 with the inception of world wide web (internet). In the inception phase of WWW the information mostly did correspond of text which was stored on different servers and was retrieved all across globe by different people using application software known as browsers. The information available on a web page was accessed using a browser which comprised of text and links and were navigated from one web-page to the other. The collection of web-pages was known as websites of a particular entity and the link was known as hyperlink. Initially all these websites were used to share information but now it has transferred into an environment wherein it has become combination of many different types of applications like an online shopping portals, travel information portals, ticketing portal, social networking or a large-scale commercial enterprise-planning and scheduling portal. The definition of web application has changed over time from a simple information sharing website to a complete one stop solution for a particular domain or a group of services. At present numerous web application are fully functional system that provide B2C, B2B, C2C services to users all across the globe [1][2].

The concept of web applications has revolutionized the definition of business from physical outlets to online outlets which has resulted in huge demand for development of web application for different business enterprises. The revolution of internet with the help of technology has removed all the borders which have resulted in a competitive software market where whole world competes to get a customer. The demand of web application development doesn't end with the customers from Industry as the governments are also using the services of internet to provide information and services to the citizens of their respective countries. The rapid spread of web applications into the areas of communication and commerce makes it one of the leading and most important branches of the software Industry [3].

The rapid increase in the demand for web applications has led to increase in number of software development enterprises providing web solutions, therefore paving way to more skilled manpower in this domain. To develop a web application a software development house has to ensure that the development is completed on time and within budget. In order to ensure this different organisations use different approaches to ensure cost over-run or time over-run. Software effort estimation for web application is a process of approximating the required effort to develop a web application. The most critical activity in any software project management is approximation of effort early in the software development life cycle. The importance of effort estimation has been justified by many researchers in the literature. A report by cost cutter consortium in 2000 shows some alarming statistics [1] which was derived from a large database of effort estimation related to web-based projects and are as given below:

79% of the projects studied presented schedule delays;
63% of the projects studied exceeded budgets;
84% of the projects studied did not meet requirements;
53% of the projects studied did not provided required function; and
52% of the projects studied had poor quality of deliverables.

The most common issue the software development industry encounters is exceeding budgets, schedule delays and not meeting requirements. The accurate estimate for a software system early in the study phase of software development life cycle plays a crucial role in software project management.

In order to manage a software development process efficiently an accurate effort estimation method is very important. However, the question such as "How and what can be optimal and correct method for effort estimation" has always been a matter of much deliberations among the researchers and practitioners.

2. SOFTWARE EFFORT PREDICTION PROCESS

Realistic effort estimates are fundamental for the successful management of software projects; the Web applications are no exception. Having realistic estimates at an early stage in a project's life cycle allows project managers and development organisations to manage their resources effectively. To this end, prediction is a necessary part of an effective process, whether it is authoring, design, testing, or Web development as a whole. A prediction process involves:

- The identification of measures (e.g. number of new Web pages, number of new images) that are believed to influence the effort required to develop a new Web application.
- The formulation of theories about the relationship between the selected measures and effort (e.g. the greater the number of new static Web pages, the greater the development effort for a new application).
- The capturing of historical data (e.g. size and actual effort) about past Web projects or even past development phases within the same project.
- The use of this historical data to develop effort estimation models for use in predicting effort for new Web projects.
- The assessment of how effective those effort estimation models are, i.e. the assessment of their prediction accuracy.

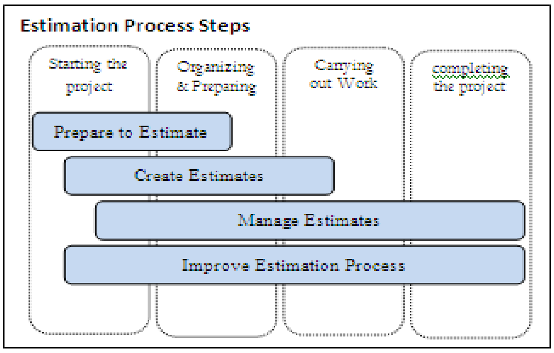

Figure 1: Estimation Process Steps

- Cost and effort are often used interchangeably within the context of effort estimation (prediction) since effort is taken as the main component of project costs. Different effort estimation techniques have been proposed and compared over the last 20 years, however, a good effort estimation technique must provide an effort estimate which nearer to the *actual* effort of the same software development project [4]. The effectiveness of effort estimation techniques to provide accurate effort estimates is called prediction power.

2.1 Effort Estimation Techniques

The purpose of estimating effort is to predict the amount of effort to accomplish a given task, based on knowledge of other project characteristics that are believed to be related to effort. Project characteristics (i.e. independent variables) are the input, and effort (dependent variable) is the output we wish to predict as shown in figure 2.

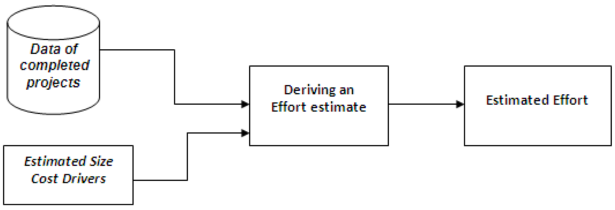
Data of completed projects
Estimated Size Cost Drivers
Deriving an Effort estimate
Estimated Effort

Figure 2: Components of Estimation Model

A task to be estimated can be as simple as developing a single function (e.g. creating a table on the database) or as complex as developing a large application, and in general the one input (independent variable) assumed to have the strongest influence on effort is size. Other independent variables may also be influential (e.g. developers' average experience, number of Web Effort Estimation tools employed) and these are often identified as cost drivers. Depending on the techniques employed, we can also use data on past finished projects to help estimate effort for new projects. Several techniques for effort estimation have been proposed over the past decades in software engineering and have been categorized into three categories [5] as expert opinion, algorithmic models and artificial intelligence techniques.

2.2.1 Expert Opinion

Expert opinion represents the process of estimating effort by subjective means, and is often based on previous experience from developing/ managing similar projects. It has been and still is widely used in software and Web development. The drawback of this technique is that it is very difficult to quantify and to determine those factors that have been used to derive an estimate, making it difficult to repeat.

2.2.2 Algorithmic Techniques

The most popular techniques described in the effort estimation literature are algorithmic techniques. Such techniques attempt to formalise the relationship between effort and one or more project characteristics. The result is an algorithmic model. The central project characteristic used in such a model is usually taken to be some notion of software size (e.g. the number of lines of source code, number of Web pages, number of links). This formalisation is often translated as an equation:

$$\textit{Estimated Effort} = a\ \textit{EstSizeNewproj}^{\,b}\,\textit{EAF} \qquad (1)$$

where *a* and *b* are parameters that also need to be estimated. Equation (1) shows that size is the main factor contributing to effort, and can be adjusted according to an Effort Adjustment Factor (*EAF*), calculated from cost drivers (e.g. developers, experience, tools). An example of an algorithmic model that uses equation (1) is the Constructive COst MOdel (COCOMO) model [6] .

Regression-based algorithmic models are most suitable to local circumstances such as “in-house” analysis as they are derived from past data that often represents projects from the company itself. Regression analysis is used to generate regression-based algorithmic models, provides a procedure for determining the “best” straight-line fit to a set of project data that represents the relationship between effort (the response or dependent variable) and project characteristics (e.g. size, experience, tools, the predictor or independent variables).

Regarding the regression analysis itself, two of the most widely used techniques are multiple regression (MR) and stepwise regression (SWR). The difference between both is that MR obtains a regression line using all the independent variables at the same time, whereas SWR is a technique that examines different combinations of independent variables, looking for the best grouping to explain the greatest amount of variation in effort. Both use least squares regression, where the regression line selected is the one that reflects the minimum values of the sum of the squared errors. Errors are calculated as the difference between actual and estimated effort and are known as the residuals.

2.2.3 Artificial Intelligence Techniques

Artificial intelligence techniques have, in the last decade, been used as a complement to, or as an alternative to, the two categories discussed above. Some of the examples of AI technique are fuzzy logic, regression trees, neural networks and case-based reasoning.

2.2.4 Measuring Effort Prediction Power and Accuracy

An effort estimation model *m* uses historical data of finished projects to predict the effort of a new project. Some believe this is enough to provide accurate effort estimates. However, to gauge the accuracy of this model we need to measure its predictive accuracy.

To measure a model's predictive accuracy first calculate the predictive power for each of a set of new projects P_1 to P_n that used the effort estimation model *m*. Once predictive power for P_1 to P_n has been obtained, their values are aggregated, which gives the predictive power of model *m* and hence it's corresponding predictive accuracy.

2.2.5 Measuring Predictive Power

The most common approaches for measuring predictive power of effort estimation models [7] are:

i. The Mean Magnitude of Relative Error (MMRE)

ii. The Median Magnitude of Relative Error (MdMRE)

iii. The Prediction at level *n* (Pred(*n*))

In order to validate a cost estimation model the predictive accuracy of a given effort estimation model needs to be calculated.

3. MOTIVATION

In an era of globalization where everyone from an organization to an individual wants to communicate with others about achievements, identity, perception and what not through a platform provided by world wide web (WWW). As the communication channel using internet has become affordable therefore, it has empowered people with the access to information anytime and anywhere resulting in bridging the gap between sender and the receiver. One can imagine a virtual world where one can access information anytime using internet and appropriate software.

Every individual at present is increasingly relying on software either directly or indirectly resulting in huge scope for web application development. There is seemingly exigent demand for more functionality, interfaces easier to use, faster to respond, lesser defects, adaptable to changing environment and flexible. Web application developers must strive to achieve these objectives while simultaneously reducing development cost and time. The managers of software development houses want to deliver the software solutions as per the deadline for which they have to manage the whole process of development from inception till maturity. To develop such applications Web development teams need to use sound methodologies, systematic techniques, quality assurance, rigorous, disciplined and repeatable processes, better tools, and baselines. A study was reported in 2009 by Standish group where 9000 projects were surveyed out of which 32% were delivered with full capability within budget & schedule, 29% were cancelled and 44% were either over budget or over-schedule and 24% were undelivered due to poor software estimates.

The ability of project managers to plan a project and ensure efficient development depends on accurate effort estimates. The importance of accurate effort estimates is illustrated by the findings of a 2007 survey of more than one thousand IT professionals. The survey reports that two out of the three-most-important causes of IT project failure were related to poor effort estimation, that is, inaccurate effort estimates. A review of estimation accuracy studies [8] reports that software projects expend, on average, 30%-40% more effort than is estimated.

The most common bottle-neck that web application development industry faces is effort estimates for a web application. Software effort and schedule estimation supports the planning and tracking of software process. The software effort/cost estimates help a web developer to ensure that the process or a function is developed as per the deadline and within budget. The studies mentioned above add to the understanding of the importance of effort estimation for web application in early stages of software development life cycle. This provides the motivation to consider the early prediction of efforts estimation of web application.

4. LITERATURE REVIEW

The study of different research papers regarding effort estimation, comparison for web application and related domain from 2000 to 2014 is reviewed as under:

E. Mendes, et. al. [9]

Findings:

- Case-based reasoning showed the better results after comparing case-based reasoning, linear regression, and stepwise regression techniques

Gap:

- The size measures used during the study are not enough to justify a best performed technique for effort estimation.

D. J. Reifer, [10]

Findings:

- Proposed a Web cost estimation model;
- WEBMO extended from COCOMO-II model.

Gap:

- No information is given regarding the data collection or any summary statistics for the data

E. Mendes, et. al. [11]

Findings:

- Different measures of prediction accuracy gave different results.
- MMRE and MdMRE showed better prediction accuracy for multiple-regression models.
- CBR (Case-Based Reasoning) showed the better results after comparing CBR, linear regression, and stepwise regression techniques

Gap:

- Contrary to the expected, late measures did not show statistically significant better predictions.

M. Ruhe, et. al. [12]

Findings:

- This result significantly outperformed expert estimates from Allette Systems (MMRE 0.37).
- A result comparable to Web-COBRA was obtained when applying ordinary least squares regression with size in terms of Web Objects as an independent variable (MMRE 0.23)

Gap:

- Many start-up companies developing web applications suffer from underestimating the cost of a new project.
- Ad-hoc approaches that could be applied in the beginning of web development do not work for larger web applications.

Baresi, L [13]

Findings:

- They investigated the impact of a few attributes (e.g., size, complexity) of the W2000 design artifacts built during the design phase on the total effort needed to design web applications
- They identified a few attributes that may be related to the total design effort.

Gap:

- The empirical study is an exploratory study
- The study needs to be revisited and checked with data for accuracy.

D. Jensen, et. al. [14]

Findings:

- underestimated project cost can bring the project with many quality-threatening “cost-cutting” measures even “project cancellation”.
- Overestimated software cost will lose the bid, waste time, money, staff and other resources, lead to the financial loss, even bankruptcy of the organization.

Gap:

- Not able to accurately estimate the software cost needed to complete the project on time is crucial for software organizations.

A. Marchetto [15]

Findings:

- A metrics suite to measure Web software modelled via Object-Oriented techniques;
- The proposed suite is useful to analyze software designed via feature concerns approach, based on Multi-Dimensional Separation of Concerns (MDSOC).
- The suite is composed of traditional OO metrics and by new metrics tailored to Web software.

Gap:

- The measurement approach was developed in the context of WAAT (Web Applications Analysis and Testing) project

S. D. Martino, et. al. [16]

Findings:

- The issue of effort estimation for the development of Web portals;
- Categorized effort estimation as non-model based estimation and model-based estimation..

Gap:

- Details about the validation methods were not given.

F. Ferrucci, et. al. [17]

Findings:

- The results provided reveal the effectiveness of Web-COBRA in combination with COSMIC.

Gap:

- Which domains can web-COBRA and COSMIC be used;
- Which activity COSMIC model should be used for estimation.

A. Idri, et. al. [18]

Findings:

- The results showed that the use of an optimal number of fuzzy sets improves greatly the estimates generated by fuzzy ID3 model.

Gap:

- data set used was only from tukutuku software project dataset

F. Ferrucci, et. al. [19]

Findings:

- Proposed Tabu Search which is a meta-heuristic approach successfully used to address optimization problems in several contexts.
- The results are encouraging given that Tabu Search outperformed all the other estimation techniques against Linear Regression and Case-Based Reasoning.

Gap:

- The study has been carried out using tukutuku data;
- Data from Industry needs to be used for validating the performance of Tabu Search.

L. Lavazza, at. al. [35]

Findings:

- Their results indicated that, overall, COSMIC Function Points are better suited than traditional Function Points for measuring characteristic features of real-time and embedded systems.

Gap:

- Neither FPA nor the COSMIC method accounts for the complexity;

C. Lokan, et. al. [11]

Findings:

- Their results suggested that it is not always beneficial to exclude old data when estimating effort for new projects.

Gap:

- Non-availability of recent data related to completed projects

Based on the findings of different researchers, there is neither well-defined metrics model able to measure the development time of a Web Application nor is it feasible to adapt the most well-known models (Cocomo-II [22] and Function Point Analysis [23]) to the domain of these software applications. The need for further empirical investigations is the motivation to undertake this research.

5. CONCLUSION

One of the crucial aspects of Software Project Planning and Management is to understand what will be the approximate cost/effort required to develop a software project without compromising on quality. The cost overruns can cause customers to cancel projects whereas cost underestimates can force a project team to invest much of its time without financial compensation. To compute accurate estimates for web applications, the software engineers have developed techniques for developing relationships among effort, staff characteristics and other factors affecting time, effort & cost of developing a software/ web application.

In order to get accurate estimation results for web-applications, a software cost estimator should carry good cost estimation methods. Proper selection of cost/effort estimation technique, correct size measures, person experience and familiarity of software developed can be of great help to estimate accurate estimates for web-applications. It is not necessary to stick to one and only one model, rather the estimation procedure should be flexible to allow more than one techniques to be used in tandem for accurate efforts. The need of the hour is to have a model which can reduce the gap between approximate cost and actual cost of a web-application. Second, the effort/cost estimation model for web-application should help in approximating better estimates in the early stage of software development life cycle to allow project leaders/ managers to monitor and manage the development process effectively and efficiently. However, it is very important to keep all the details related to the approximated estimates and actual estimates transparent which will result in management of project and schedule at individual level.

6. REFERENCES

[1] E. Mendes, N. Mosley, and S. Counsell, “Do adaptation rules improve Web cost estimation,” Proceedings of the ACM Hypertext Conference, pp.173-183, 2003.

[2] L. Baresi, S. Morasca, And P. Paolini, “Estimating the design effort of Web applications”, Proceedings of the Ninth International Software Measures Symposium, pp. 62-72, 2003.

[3] B. Davide, and P Paolini, “Goal-driven requirements analysis for hypermedia-intensive Web applications.” Requirements Engineering vol. 9, no. 2, pp. 85-103, 2004.

[4] D. Kashyap, and A. K. Misra, “Software development cost estimation using similarity difference between software attributes.” In Proceedings of the International Conference on Information Systems and Design of Communication ACM, pp. 1-6, 2013.

[5] M. J. Shepperd, and C. Schofield, “Estimating software Project Effort using Analogies.” Software Engineering, IEEE Transactions, vol.23 no. 11, pp. 736-743, 1997.

[6] B. Clark, S. Chulani, and B. Boehm, “Calibration Results of COCOMOII.1997”, In International Conference on Software Engineering, pp. 477-480, 1998.

[7] E. Mendes, S. Counsell, N. Mosley, C. Triggs, I.Watson, “A Comparative Study of Cost Estimation Models for Web Hypermedia Applications”, Empirical Software Engineering vol. 8 no. 2, pp. 163-196, 2003.

[8] K. Moløkken, and M. Jørgensen, “A review of software surveys on software effort estimation.” Proceedings of International Symposium on Empirical Software Engineering, Rome, Italy: Simula Research Lab. Lysaker Norway, pp. 223-230, 2003.

[9] E. Mendes, S. Counsell, and N. Mosley, “Measurement and effort prediction of Web-applications,” Proceedings of the 2nd ICSE Workshop on Web Engineering, pp. 57-74, 2000.

[10] D. J. Reifer, “Web Development : Estimating Quick-to-market software”, IEEE software, vol. 17 no. 6, pp. 57-64, 2000.

[11] E. Mendes, N. Mosley, and S. Counsell, “The application of case-based reasoning to early Web project cost estimation,” Proceedings of IEEE COMPSAC, pp. 393-398, 2002.

[12] M. Ruhe, R. Jeffery, and I. Wieczorek, “Cost estimation for Web applications,” Proceedings of 25th International Conference on Software Engineering, IEEE, pp. 285-294, 2003.

[13] Baresi, L.; Morasca, Sandro; Paolini, P., "Estimating the design effort of Web applications," Software Metrics Symposium, 2003. Proceedings. Ninth International, pp.62,72, 2003

[14] D. Jensen, and W. Randall, “Extreme Software Cost Estimating”, The Journal of Defense Software Engineering vol.17, no. 1, 2004.

[15] A. Marchetto, “A concerns-based metrics suite for web applications.” INFOCOMP journal of computer science, vol. 4 no. 3, 2004.

[16] S. D. Martino, F. Ferucci, and C. Gravino, “Effort Estimation for the Development of Web Portals.” Encyclopaedia of Portal Technologies and Applications, IGI Global, pp. 304-309, 2007.

[17] F. Ferrucci, C. Gravino, and S. Di. Martino, “Estimating Web Application Development Effort Using Web-COBRA and COSMIC: An Empirical Study.” In Software Engineering and Advanced Applications, SEAA'09. 35th Euromicro Conference on. IEEE. 2009.

[18] A. Idri, and S. Elyassami, “A Fuzzy Decision Tree to Estimate Development Effort for Web Applications.” International Journal, 2011.

[19] F. Ferrucci, C. Gravino, R. Oliveto, F. Sarro, and E. Mendes, “Investigating Tabu Search for Web Effort Estimation.” In Software Engineering and Advanced Applications (SEAA), 36th EUROMICRO Conference on IEEE, pp. 350-357, 2010.

[20] L. Lavazza, and S. Morasca, “Measuring the Functional Size of Real-Time and Embedded Software: a Comparison of Function Point Analysis and COSMIC.” In ICSEA, The Eighth International Conference on Software Engineering Advances, pp. 465-470, 2013.

[21] C. Lokan, and E. Mendes, “Investigating the use of duration-based moving windows to improve software effort prediction: A replicated study.” Information and Software Technology Elsevier, 2014.

[22] B. Boehm, C. Abts, B. Clark, and S. Devnani-Chulani, "COCOMO II model definition manual," University of Southern California, 1997.

[23] A. J. Albrecht, and J. E. Gaffney, “Software function, source line of code and development effort prediction: A software science validation,” IEEE Transaction on Software Engineering, vol. 9, no. 2, pp. 639-648, 1983.

© 2015-19, IJARCS All Rights Reserved