Global Journals LaTeX JournalKaleidoscope™
Artificial Intelligence formulated this projection for compatibility purposes from the original article published at Global Journals. However, this technology is currently in beta. *Therefore, kindly ignore odd layouts, missed formulae, text, tables, or figures.*

# An Approach for Effort Estimation having Reusable Components in Software Development

Jyoti Mahajan[1]

1

*Received: 3 October 2011 Accepted: 31 October 2011 Published: 14 November 2011*

## Abstract

Estimation of the effort required for development has been researched for over 25 years now. Still there exists no concrete solution to estimate the development effort. Prior experience in similar type of projects is a key for business today. This paper proposes an Effort Estimation Model named REBEE based on the reusable matrices to effectively estimate the effort to be involved for development. A project is assumed to consist of multiple modules and the reusability factor of each module is considered in the technique described here. REBEE utilizes fuzzy logic and dynamic neural networks to achieve its goal. Based on the experimental evaluation discussed in this paper it is evident that this model accurately predicts the effort involved on heterogeneous project types.

***Index terms*** — Software Effort Estimation, Software Reusability, Dynamic Neural Networks, Fuzzy Logic, REBEE.

## 1 INTRODUCTION

OFTWARE EFFORT ESTIMATION is crucial to derive the effort involved in the successful completion of any project. Effort estimation techniques facilitate financial estimates, delivery timelines, help in beneficial resource allocation and scheduling, monitoring progress and also help in risk management. According to a recent survey conducted by McKinsey for NASSCOM ??1] the IT and allied industries are expected to bring in revenues of about $225 Billion by 2020 in India alone and the current revenues are about $76 Billion. It is evident from these figures the growth rate of the software industry is impressive. The recent years have observed that software contracts are awarded to organizations having prior experience in handling similar project types.

Prior experience in the related project is the key for business growth. Organization benefiting from the software contracts would have multiple reusable modules for their future work. More over organizations develop codes so that they could be reused with some modifications for future use. This conservative approach adopted by the industry is to ensure timely deliveries, quality, reliability and financial assurance of their investments.

COCOMO [3] and COCOMO 2.0 [4], DELPHI [5], Function Point [6], Planning Poker [7], Use Case Point [8], Expert judgment [9], IBM -FSD [10] are the world known based estimation techniques, which are commonly used for Software Effort Estimation. These models exhibited a gross error of effort estimation. COCOMO with effort adjustment factor [11] provides about 30% improvement in effort variance, whereas when it is used with fuzzy logic, trapezoidal function and Gaussian functions showed improved performance [12]. Multiple software effort estimation techniques were integrated together to get the better result as compare to the regularly used estimation techniques, which was the big failure in terms of consistency when tested against several cases.

It was found that to achieve the good accuracy, Support Vector regression was combined with clustering approach.

The estimation algorithm was vastly improved by the Mantel's correlation randomization test named Analogy-X [15]. This made the researchers to work even harder on the after effects of Schedule and Budget pressure on Effort Estimation and the development cycle time. Researchers have to be very careful while Chronological

Splits are assigned for the testing and training purpose. Even Global Software Developments gets an inaccurate estimation technique being executed in different location of all over the world.

It has become very difficult to decide which model like COCOMO is best suitable for the development of the estimation model because of the different efforts to achieve estimation technique available in the market and the same outputs. The best solution for the estimation technique can be the judgment and the formal based model. In spite of all these available models and approaches, research shows the failure of projects due to various reasons [13]. Project Failures due to improper estimation techniques is also studied [14]. Based on this study it is evident that appropriate effort estimation techniques are critical for project success. The current existing techniques provide no proper estimation and are not applicable for varied project types.

To estimate effort for heterogeneous project types this paper discusses REBEE in the further sections of the paper. The remaining paper is organized as follows. The next section discusses the importance of reusability and its adoption in the industry today. The third section discusses the REBEE model proposed. Section 3 also presents the Fuzzy rules to derive the reusability matrix and its use with dynamic neural networks to estimate effort. The penultimate section presents the experimental evaluation conducted using REBEE. The conclusion of the research presented here is discussed in the last section.

# 2 II.

# 3 REUSABLITY AND ITS IMPORTANCE

The software industry today has witnessed various changes in its formulation, maintenance and management strategies to adapt to the dynamic changes it has experienced and for greater profitability. Experience held with organization in relevant or similar projects provides them with an business advantage as discussed earlier. These organizations possess modules which could be altered or used in total for their upcoming projects. The work described in this paper utilizes this knowledge of these reusable components to predict the effort required for the remaining work at hand. Incorporation and importance of reusability is currently been actively considered by major corporations now. Reusability is being considered for appraisals of employees of an organization [16] to reduce costs and maximizing profits [17]. Through these studies it is evident the adoption and importance of reusable components in the industry today and effort estimation using based on reusability could answer the anomalies that exist in the current estimation techniques adopted.

Fellow researchers have incorporated reusable weights into the existing COSYMO for cost estimation [18].Incorporation of the reusable parameters with the taguchi model [19], COCOMO2 [20],COCOMO [11] and COCOMO81 [20] have been closely observed and these models exhibit considerable improvements but the error of estimation still exist. The error in estimation is basically due to the fact that the deficiencies of reusability's were not considered [21] which was considered to develop REBEE. The effort estimation technique proposed consists of a pre processing phase where in the project data considered is analyzed to basically derive the reusability matrix. A project is assumed to be split into a number of modules and the reusability of each module is analyzed to derive the reusability matrix using fuzzy rules.

Estimation of the effort involved to achieve the project goals have been achieved using dynamic neural networks. Prior to estimation the dynamic neural networks are trained using the back propagation algorithm. The trained neural network could be used for estimation the effort involved. The results obtained could be analyzed for resource utilization, financial analysis, delivery time line assertion and many more critical analyses.

# 4 b) Reusability Matrix using Fuzzy Logic

A project is said to be composed of ?? modules. Modules could be either reusable or could be considered as new modules (?? ?? ). Each reusable module is analyzed using a judgment model to arrive at the reusable component present. The modules are analyzed at an implementation level and for characterization a threshold ?? is defined which is arrived based on the judgment model. On characterization the modules are further classified into 3 categories as? Completely reusable.

A module is considered to be completely reusable if it could be utilized without any changes or changes to be incorporated are less than the threshold ?? and is represented as ?? ð ??”ð ??”?? ? Reusable with prominent adaptation

If the changes to be incorporated are greater than the threshold ?? then the module is considered as a reusable module with prominent adaption represented by ?? ???? .

Let Î?” represent the changes to be incorporated into a module ?? for it to be compatible with the project for which estimation is to be achieved. Applying the fuzzy rules the modules could be characterized as follows? ?? = ?? ???? ??ð ??”ð ??” ?????? ???????? ??ð ??”ð ??” Î?” = 0 ?? = ?? ð ??”ð ??”?? ??ð ??”ð ??” ?????? ???????? ??ð ??”ð ??” Î?” < ?? ?? = ?? ð ??”ð ??”?? ??ð ??”ð ??” ?????? ???????? ??ð ??”ð ??” Î?” ? ?? ?

Consider ?? to represent the reusable matrix. The effort involved to develop the modules earlier is represented as ? . Let us consider that there exist ??, ?? ?????? ?? number of ?? ???? , ?? ð ??”ð ??”?? ?????? ?? ???? modules and their development efforts considered be defined as ? ???? , ? ð ??”ð ??”?? ?????? ? ???? . Then the reusability matrix obtained based on fuzzy logic could be represented as Dynamic neural networks have been considered as they could be utilized to observe effort related dynamics of the input pattern matrices. The use of dynamic neural networks is not only related to obtaining effort related dynamics but also could be utilized to obtain non effort related dynamics observed for effort related input matrices.?? = ? ?? ????1 ? ????1 ?? ð ??”ð ??”?? 1 ? ? ? ?? ?????? ? ?????? ?? ð ??”ð ??”???? ? ð ??”ð ??”?? 1 ?? ???? 1 ? ???? 1 ? ? ? ? ð ??”ð ??”???? ?? ?????? ? ?????? ? c)

The reusable matrix obtained from the pre processing phase is considered for training of the dynamic neural networks. The training is achieved using the back propagation algorithm. The output of the dynamic neural network ??(??) with respect to the input ??(??) is given by??(?? + 1) = ?(?? ? 1)??(??) + ??????(??) + ?? ??(??) = ?????(??)?

Where ?? represents the sigmoid activation function and (?? ? 1) is the feedback where ?? is the learning rate constant.

The error of estimation ??(??) is defined as??(??) = 1 2 ??? ?? (??) ? ??(??)? 2 + 1 2 ? [?? ?? (??) ? ??(??)] 2 ???1 ??=0 = 1 2 ?? 2 (??) + 1 2 ? ?? 2 ???1 ??=0 (??)

The weight update function ?????(??) propagated through the dynamic neural network is given as?????(??) = ??? ???? ?????? = ?? ? ??(?? + 1)ð ??”ð ??” ???? (??(??), ????) ???1 ??=0

The updated weights propagated to the next neuron based on the previous neuron is given as????(?? + 1) = {????(??) ? ?? ???? ? ??(?? + 1) ???1 ??+0 ð ??”ð ??” ???? (??(??), ????)}

The trained neural network is queried with the project data provided which provides the effort estimated on the remaining modules using the following equation where ? represents the effort.

# 5 ??(?? + 1) = ?(??? ? 1)??(??) + ?ð ??”ð ??”(??(??), ????) + ???(??)

This section of the paper described the REBEE technique proposed through this paper. The validation of this model is provided in the next section.

# 6 IV. EXPERIMENTAL VALIDATION OF THE REBEE TECHNIQUE

This section of the paper would discuss the experimental evaluation of the discussed REBEE model. For evaluation 39 projects of NASA Goddard Space Flight Center Greenbelt, Maryland is considered [22]. The dataset consists of projects related to simulators and altitude ground support systems developed by the Flight Dynamics Division of Goddard Space Flight Center situated in Maryland USA. The simulator projects considered were categorized into dynamic simulators and telemetry simulators. The 39 projects considered were said to be developed in 3 phases. Phase 1 consist of the design Phase. The coding was considered as the second phase and the last phase was the testing phase. For evaluation purpose the effort involved in providing support towards these projects developed was not considered.

# 7 Figure 3 : Experimental Evaluation Flow Diagram

The data set considered provided details with respect to the number of lines of source code required in developing these projects. Reusability of the code was also considered in the development of the projects in the data set. The data set defined reusability of 3 types. A completely reusable code was considered if there were no changes to be incorporated for the new project considered. If the changes to be incorporated in the code were less than 25% (i.e. threshold ?? in REBEE) then it was considered to be a reusable code that requires slight modification. If the modification exceeded the threshold ?? the code was considered to be reusable but with extensive modification. For evaluation presented here these matrices were considered to derive the reusability matrix??.

REBEE was developed using C# on a visual Microsoft Visual Studio 2010 platform. The reusability matrix derived using the fuzzy rule was provided to the dynamic neural network in the training process. The trained dynamic neural network on querying provides the effort estimated phase wise and for the entire project.

The experimental evaluation process considered is shown as a flow chart in Fig 3 **??**The dataset considered consists of heterogeneous project having varied development platforms and also exhibiting varying reusability levels. For evaluation projects were clustered into 4 types mentioned below? Minor Reuse ? Standard Reuse ? High Reuse ? Maximum Reuse types.

If the reusability of a project was found to be less than or equal to 20% it was considered to be of Minor Reuse Type. If the reusability percentage of a project was between twenty and fifty, it was considered as a project of Standard Reuse type. If the percentage of reusability of a project was between fifty and eighty it was considered as a project of High Reuse. Projects embodying components which were more than 80% reusable was considered as maximum reusable projects. This clustering was adopted to provide for effective and efficient training to neural network to understand the dynamics of reusability. The effort estimated versus the actual effort involved in the design phase of the 39 projects is shown in The results obtained from the evaluation of the 39 projects considered exhibited a low average error in effort estimation of about 1.25%. The average error in An Approach for Effort Estimation having Reusable Components in Software Development effort estimation for the design phase was 1.34%, 1.38% for the coding phase and 2.29% for the testing phase respectively. Based on the graphical data provided and the low average error of estimation it is evident that the reusability based effort estimation technique presented in this paper could be effectively utilized to estimate the effort involved in developing a project.

# 8 CONCLUDING REMARKS

Accurate effort estimation techniques are critical for the successful project execution. The importance of reusability and its remarkable acceptance by the industry today is evident from the research work presented through this paper. This paper discusses a reusability based effort estimation technique named REBEE. Training of the dynamic neural networks is achieved using the back propagation algorithm. Fuzzy rules are adopted in constructing the reusability matrix which is utilized by the neural network to understand the dynamics of the effort involved in constructing the reusable components. Based on this understanding the dynamic neural network estimates the remaining effort involved in project completion.

The REBEE model discussed is evaluated on 39 NASA projects which are of different kinds. The development languages for these projects also varied from project to project. The reusability level of the projects varied from about 0% to a high of 96%. The effort estimated using REBEE on all the 3 project phases i.e. Design, Coding and Testing and on the cumulative effort required in developing the projects showed high levels of accuracy. The average estimation error for all the 39 projects was also a low of about 1.25% which proves the efficiency of REBEE. From the evaluation results obtained it could be concluded that reusability based effort estimation technique discussed in this paper could be a possible solution for accurate effort estimation for projects of varied types which is not possible with the currently existing effort estimation techniques. [1] [2] [3] [4] [5] [6]

Figure 1:

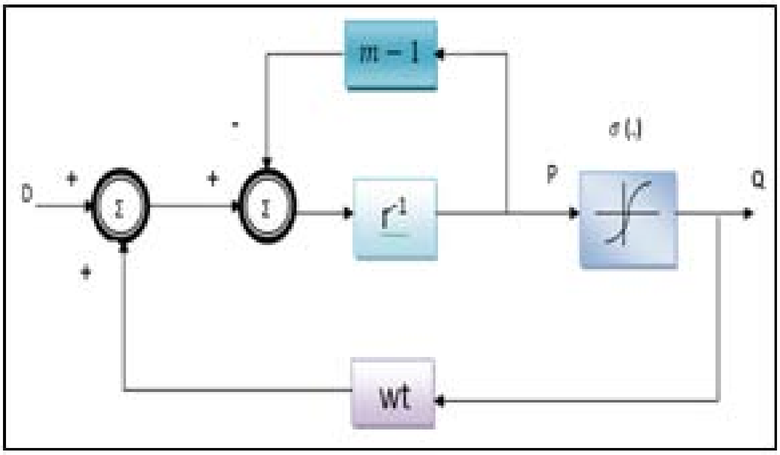


Figure 2: Figure

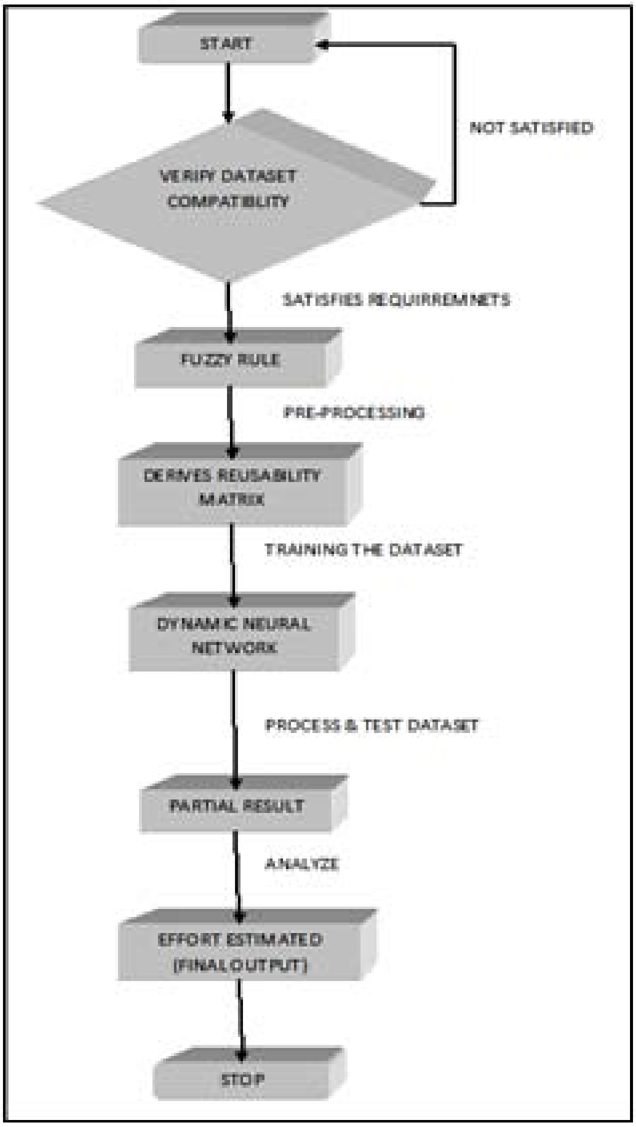


Figure 3:

Figure 4:

90000 Complete Project Effort Estimation in Man Hours 50000 Coding Phase -
100000 45000
80000 REBEE Estimated Total Effort 40000
70000 35000
60000 Actual Total Effort 30000
50000 25000
40000 20000
30000 15000
20000 10000
10000 5000
0 1 6 11 16 21 26 31 36 0 1 6 1
Design Phase -Effort Estimation in Man 30000 Tesing Phase -E
25000 Hours
20000 REBEE Estimated Design Effort 25000
Actual Design Phase Effort
15000 20000
10000 15000
5000
10000
0
1 6 11 16 21 26 31 36 5000
0
1 6 1

Figure 5:

[1] © 2011 Global Journals Inc. (US)

[2] December

[3] December

[4] © 2011 Global Journals Inc. (US) Global Journal of Computer Science and Technology Volume XI Issue XX Version I 34

[5] © 2011 Global Journals Inc. (US) Global Journal of Computer Science and Technology Volume XI Issue XX Version I

[6] © 2011 Global Journals Inc. (US) Global Journal of Computer Science and Technology Volume XI Issue XX Version I 36 2011 December

# .1 ACKNOWLEWDGEMENT

The authors would like to express their cordial thanks to Prof. Bhopinder Singh, Principal of Government College of Engineering and Technology, Jammu for their financial support and advice.

1. NASSCOM ”Perspective 2020: Transform Business, Transform India” http://www.nasscom.in/upload/Perspective%202020%20Press%20release%20pres entation.pdf

[Boehm et al. ()] , B W Boehm , W W Royce , Cocomo Le , Ada . *Genie logiciel & Systemes experts* 1989.

[Walston and Felix ()] ‘A method of programming measurement and estimation’. C E Walston , A P Felix . *IBM Systems Journal* 1997. 16 (1) .

[Molkken and Jorgensen] ‘A Review of Surveys on Software Effort Estimation’. K Molkken , M Jorgensen . *Proc. 2003 International Symposium on Empirical Software Engineering (ISESE’03)*, (2003 International Symposium on Empirical Software Engineering (ISESE’03)) p. 223.

[Sandhu et al. ()] ‘A Taguchi Approach to Investigate Impact of Factors for Reusability of Software Components’. P S Sandhu , P Blecharz , H Singh . *Journal of World Academy of Science, Engineering And Technology* 2008. 25 p. .

[Kocaguneli et al.] ‘AI-Based Models for Software Effort Estimation’. E Kocaguneli , A Tosun , A Bener . *Proc. 36th EUROMICRO Conference on Software Engineering and Advanced Applications*, (36th EUROMICRO Conference on Software Engineering and Advanced Applications) p. .

[Reddy and Raju ()] ‘An Improved Fuzzy Approach for COCOMO’s Effort Estimation using Gaussian Membership Function’. C S Reddy , Raju . *Journal of Software* 2009. 4 (5) p. .

[Sandhu and Singh ()] ‘Automatic Reusability Appraisal of Software Components using Neuro-Fuzzy Approach’. P S Sandhu , H Singh . *International Journal of Information Technology* 2006. 3 (3) p. .

[Condon et al. ()] *Cost and Schedule Estimation Study Report*, S Condon , M Regardie , M Stark . SEL-93- 002. 20771. November 1993. Goddard Space Flight Center Greenbelt, Maryland.

[Boehm ()] ‘Cost Models for Future Software Life Cycle’. B W Boehm . Processes: COCOMO2.0. *Annals of Software Engineering on Software Process and Product Measurement*, (Amsterdam) 1995.

[Peixoto et al.] ‘Effort Estimation in Global Software Development Projects: Preliminary Results from a Survey’. C E L Peixoto , J L N Audy , R Prikladnicki . *Proc. 2010 5th IEEE International Conference on Global Software Engineering, ICGSE*, (2010 5th IEEE International Conference on Global Software Engineering, ICGSE) p. .

[Basavaraj and Shet ()] ‘Empirical validation of Software development effort multipliers of Intermediate COCOMO Model’. M J Basavaraj , K Shet . *Journal of Software* MAY 2008. 3 (5) .

[Meli and Santillo] ‘Function point estimation methods: a comparative overview’. R Meli , L Santillo . *Proc. 1999*, (1999)

[Hari et al. ()] ‘Identifying the Importance of Software Reuse in COCOMO81, COCOMOII’. Ch V M K Hari , P V G D Reddy , J N V R Kumar , G Sriramganesh , V M K Ch , Hari . *International Journal of Computer Science and Engineering JCSE* 2009. 1 (3) p. .

[Ozarin (2008)] ‘Lessons Learned on Five Large-Scale System Developments’. N Ozarin . *IEEE Instrumentation & Measurement Magazine, nol* February 2008. 11 (1) p. .

[Sandhu et al. (2009)] ‘Modeling of Reusability of Object Oriented Software System’. P S Sandhu , H Kaur , A Singh . *Journal of World Academy of Science, Engineering and Technology* August 2009. (56) .

[Jørgensen (2005)] ‘Practical Guidelines for Expert-Judgment-Based Software Effort Estimation’. M Jørgensen . *IEEE Software* May-June 2005. 22 p. .

[Mahajan et al.] ‘REBEE Reusability Based Effort Estimation Technique using Dynamic Neural Network’. J Mahajan , K Devanand , Dhruve . *Global Journal of Computer Science and Technology* 11.

[Wang et al. (2010)] ‘Reuse in Systems Engineering’. G Wang , R Valerdi , J Fortune . *IEEE Systems Journal* September 2010. 4 (3) p. .

[Nageswaran (2001)] ‘Test effort estimation using use case points’. S Nageswaran . *14th International Internet Software Quality Week*, (San Francisco, California, USA) 2001. June 2001.

[The European Software Measurement Conference -Amsterdam (2008)] *The European Software Measurement Conference -Amsterdam*, October 6-8.

[Webreference] `http://www.mountaingoatsoftware.com/system/presentation/file/51/bayXP_070320_PlanningAgileProjects.pdf` *Webreference*,

[Website] `http://www.stellman-greene.com/aspm/images/ch03.pdf` *Website*,