**International Journal of Academic Research in Computer Engineering**
Volume 1, Number 1, Pages: 34-39, September 2016
www.ijarce.org
*IJARCE*

# Software Cost Estimation using Adaptive Neuro Fuzzy Inference System

Freshte Dabaghchi Mokri[1*], and Maryam Molani[2]

[1]Department of Computer Engineering, Mahabad Branch, Islamic Azad University, Mahabad, IRAN.
[2]Department of Computer Engineering, University of Applied Science and Technology Boukan Center, Boukan, IRAN.

Receive Date: 2 June 2016; Accepted Date: 28 July 2016, Published Date: 15 September 2016

*Corresponding Author: f.dabaghchi@gmail.com (F.D. Mokri)

**Abstract**
*Over the past 30 years, one of the challenges of software engineers and managers of software companies in the development of software projects is to estimate accurate cost, effort, quality and risk analysis. Although over 30 years several models presented by researchers, each of them had their strengths and weaknesses. But the need for new methods to overcome the COCOMO model still exists. During the last 20 years' models based on Artificial Intelligence have been considered more than other models by researchers which among them, neuro-fuzzy models are the latest model. The aim of this paper is to reduce errors and increase accuracy in estimating the cost and effort in software development. To achieve these goals NASA63 data collection and Adaptive Neuro Fuzzy Inference System (ANFIS) models are used and we could achieve MMRE error in proposed method to 0/0984 and the accuracy of estimate to 0/889*.

**Keywords:** *S*oftware Cost Estimation, Neuro-Fuzzy Model, COCOMO Model.

## 1. Introduction

The global industry of software development has been already mature and more complex. The industry makes use of newer tools and techniques to develop software. Then this challenge lies in careful modeling and forecasting effort of software development and then creation of planning a project development. The main goal of each software development company is to develop a new project with acceptable quality within budget and on schedule planning table [1] .The process of estimating the cost of the project should be done at the beginning of project implementation by reviewing costs of new job implementation and determining the necessities [2]. But this process won't be finished because of the change of situation till the project completion [1]. To find the best model for estimation, the accuracy of the models is essential. But it is difficult to find such a model because of finding all possible factors and using them to estimate the effort [3]. At the beginning of software development cost and effort have been forecasted and then presented to employer who intended to finance the project. In situations where timing is very long or cost is very high or both of these cases, often a customer will order the delivery time to be shorter than the estimated date and also costs are estimated to be within the lower ranges.

In industry, projects that their official estimates will be dismissed and instead of using the team capabilities, self-willed programs are executed has the highest failure rates. About 60 percent of such canceled projects never completed. The remaining 40 percent which eventually come to an end, usually their schedule delays around one year and their cost will be about 50 percent higher than the target [4]. In a report by Standish Group's Chaos, 66% of analyzed software projects have delivered to the customer with delay or cost over the predicted budget or even the worst, they failed and never reach to end [6]. One of the main reasons for such failures for developing these projects, is incorrect estimation of time and cost [7]. Therefore, evaluation of new methods for improving the accuracy of such estimations is important [5]. Researchers are constantly seeking to find new models to solve the above problems. In general, the models can be divided into two general categories algorithmic and non-algorithmic [8]. Among non-algorithmic models, ANFIS model is used for better evaluation of artificial and complex neural networks [8].

Other section of this paper is organized as follows: In Section 2, the related works will be discussed during the last two years. In Section 3, the proposed model will be discussed. In Section 4, the evaluation and results of the proposed model will be discussed and finally in Section 5, we have discussion and conclusion.

## 2. Related Works

To estimate software development effort, various types of neuro-fuzzy models have been presented. [9]. Researchers named the different types of neuro-fuzzy models from the K type to J. Neuro-Fuzzy models type A, B-Unified, B-Compensation, G and J are tested by NASA-93 and Maxwell-62 data. At the end, the result of their work have been compared with several methods of error calculation. Among error calculation methods, models of B-Compensation and J other have better results than other models. It also indicated that the neuro-fuzzy model type J using data collections of NASA93 and B-Compensation type using the Maxwell-62 data collections have better results. At the end, they concluded that the appropriate neuro-fuzzy models for data collection depends on the linearity of the data collection and number of phase rules. They stated that it is difficult to estimate the software effort in the initial stage of software development because of uncertainty of input parameters [10]. ANFIS model deals with uncertainty and provides a reliable effort estimation. The researchers use NASA-93 data collection for training and NASA63 data for testing. They considered number 2 for the number of membership functions, 20 for epoch and they regarded genfis1 command to create ANFIS. They compared ANFIS membership functions based on MMR error and they concluded that ANFIS model with Trapezoidal membership functions has less MMR error than the others. Fuzzy model from Sugeno model first type is suggested for cost estimation with three inputs and one output [11]. Their inputs have bell-shaped membership function and output has a fixed membership function. Researchers used 41 module of Lopez Martin's data set, they chose randomly 30 projects for training and 12 for testing, and at the end, have compared their proposed model with several Artificial Neural Network (ANN) models, thus they concluded that the proposed ANFIS model has less MMRE error than other ANN models.

ANFIS is used to assess the software effort [8]. The authors expressed the strengths and weaknesses of the models used by researchers over many years for estimating the software effort, and their reason for choosing ANFIS was to overcome this problem. They selected 13 projects for training and 5 projects for testing from the total 18 projects of NASA_18 data set. For ANFIS training they used ANFIS based on sugeno and Hybrid optimizing method with two membership function and with 500 repetitions.

They have compared a variety of membership functions and came to the conclusion that MMRE error of "gauss2mf" membership function and RMSE error of "gbellmf" membership function is less than others.

Ziauddin et.al. in [12] used Mamadani hybrid method based on NEFCON neuro-fuzzy model along with COCOMO II model to estimate the cost of software projects. It was done with the aim of developing ad creating a simple GUI based on cost estimation of neuro-fuzzy model and analyzing results. In this research the researchers have used the Triangular membership functions.

A new model has been suggested for software effort estimation using adaptive neuro-fuzzy approach [13]. They expressed that the reason of their research was the need for software engineers to provide new models for more accurate estimation. They used NASA_18 data set and at the end they have compared their proposed model with the help of MMRE and RMSE errors with Halstead, Bailey-Basili, Walston-Felix and Doty models. MMRE and RMSE error of the proposed model is 0/119 and 7/0731, respectively which is far less than other models. Among the other models, MMRE error and RMSE of Bailey-Basili model are less than the rest of the models.

## 2. Proposed Method

In studies during the past 30 years, various definitions of software cost estimation have been presented. Researchers in [14] have suggested that the estimation of software costs is the prediction of working hours and the number of workers needed to develop a project. In most studies the meaning of estimation is to estimate the cost and effort required to create or develop software. [15] Sometimes the cost estimation, also called

parametric estimation because its accuracy needs to understand the relationship between grades of discrete parameters that these parameters can affect the results of software projects both individually and totally. The estimation should be estimated at the beginning of the software life cycle by examining the costs of new work and determining the necessities in order to assist the project managers to better manage the project [2].
In this article, NASA63 data set was used [17]. Each project has been described by 17 features, which first 15 parameters include cost estimation drivers and the sixteen parameter includes number of lines of codes (LOC) and the last parameter includes real software development effort. All of the parameters are numerical and there is no lost one. These parameters are divided into several classes from very low to very high [17]. ANFIS is an acronym which is derived from the first letters of Adaptive Neuro-Fuzzy Inference System.
ANFIS model with the help of a series of input / output data creates a ANFIS. Parameters of membership functions of the system set through back-propagation algorithm or its hybrid with the method of the least squares. This operation of adjustment allows fuzzy systems to learn their structure from data sets. A structure similar to neural networks can be used for mapping between input and output. In fact, membership functions and their parameters used for mapping inputs and neural networks used to map outputs. Parameters of membership functions are altered during learning process. Calculation of these parameters (or setting them) is facilitated through a gradient vector. The gradient vector provides a measure for utility of modeling the ANFIS parameters. After assembling gradient vector, other optimization routines can be used to optimize the parameters and reduce errors. Usually the error is calculated using the sum of squares of errors. The modeling system used in ANFIS is similar to other techniques of system diagnosing.
In the first phase, a parameters system is used as default and then input and output data are collected in a usable form in ANFIS. Then the ANFIS can be used for training [16]. In Figure 1 a diagram of the proposed model is shown.

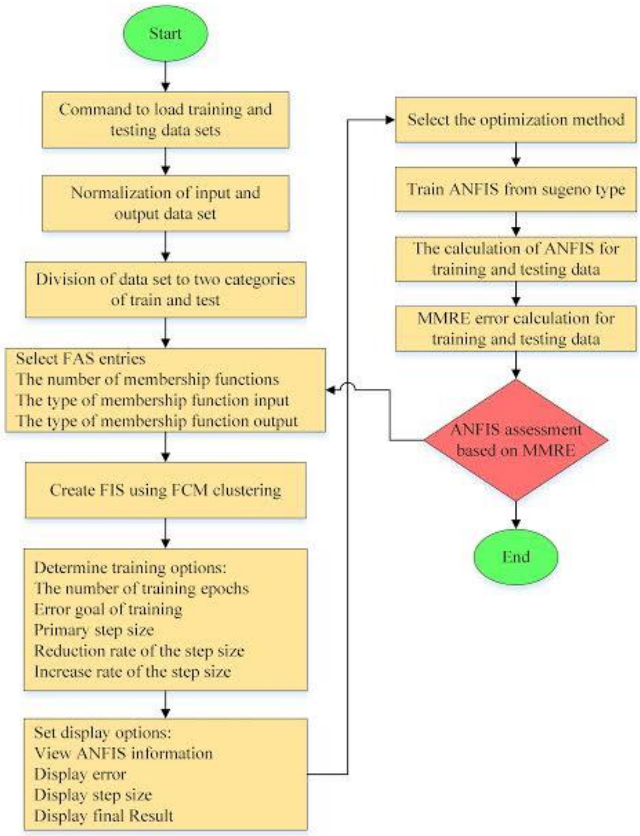


**Figure 1. Flowchart of the Proposed Method**

The most important part of dramatizing of ANN models and neuro-fuzzy is normalization. Without normalization neural network training is very slow. Data normalization increases network performance [18]. Normalization "Min_Max" is used. As it transfers the characteristics or outputs from a range of values to the new value range. Most of the range of values are between [0, 1] or between [-1, 1], which we considered them between [-1, 1]. "Min_Max" normalization formula is as the equation (1)

$$x' = \frac{x - \min_x}{\max_x - \min_x}(new\_\max_x - new\_\min_x) + new\_\min_x \quad (1)$$

From the 63 NASA63 data sets, we have randomly chosen 85% of the projects for training and 15% for testing. List of parameters used in MATLAB simulation software are summarized and are visible in Table (2).

**Table 2. Used Parameters in the Proposed Model**

| | ANFIS Method | Fuzzy C-Means Clustering |
|---|---|---|
| **ANFIS Options** | No. Input Data | 3 |
| | No. Output Data | 1 |
| | Type of ANFIS Structure | Sugeno |
| | No. Clusters | 30 |
| | Type of Membership Function of Inputs | Gaussmf |
| | Type of Membership Function of Output | Linear |
| **Training Options** | Training Epoch Number | 50 |
| | Training Error Goal | 0 |
| | Initial Step Size | 0.01 |
| | Step Size Decrease Rate | 0.9 |
| | Step Size Increase Rate | 1.1 |
| Optimization Method | | Hybrid |

Architecture of the proposed ANFIS is shown in Figure 2. It is observable that the proposed ANFIS has three inputs and one output parameters. Rule number and the number of membership functions of the inputs and outputs is equal to the number of clusters. We have considered the number of clusters 30 in this simulation. In every rule the logic operation ‘and’ has been used.

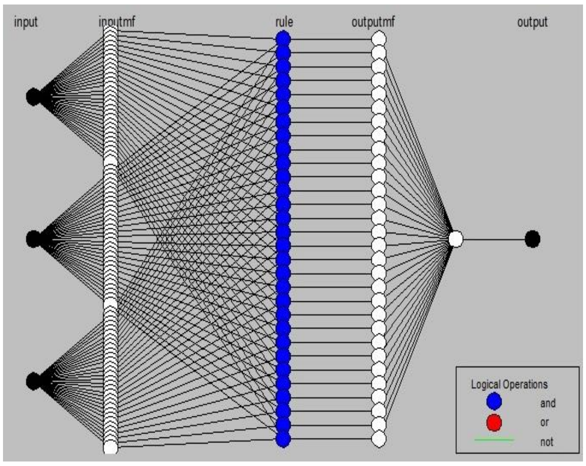


**Figure 2. The Structure of the Proposed ANFIS**

In this paper, for ANFIS simulation MATLAB software was used. The maximum number of ANFIS input parameters should be 3 parameters, but the number of involved properties in the estimated cost is 16 software features So we should find three important features for software cost estimation from these 16 features. LOC is one of the most important aspects involved in cost estimation software. So we consider one of the three inputs of ANFIS as LOC and for the other two inputs we use trial and error method. The results of this trial and error is visible in the Tables (3) and (4).

**Table 3. MMRE Calculated for Various ANFIS Input Parameters**

| Fuzzy Membership Function | ACAP, PCAP, LOC | PCAP, AEXP, LOC | AEXP, MODP, LOC | MODP, TOOL, LOC | TOOL, VEXP, LOC | VEXP, LEXP, LOC | LEXP, SCED, LOC |
|---|---|---|---|---|---|---|---|
| Train | 4.9080e-4 | 0.0063 | 6.9843e-4 | 0.0011 | 7.3624e-4 | 0.0015 | 0.0014 |
| Test | 0.0664 | 0.1298 | 1.2253 | 0.4832 | 0.2707 | 0.3498 | 0.1810 |

**Table 4. MMRE Calculated for Various ANFIS Input Parameters**

| Fuzzy Membership function | SCED, STOR, LOC | STOR, DATA, LOC | DATA, TIME, LOC | TIME, TURN, LOC | TURN, VIRT, LOC | VIRT, CPLX, LOC | CPLX, RELY, LOC |
|---|---|---|---|---|---|---|---|
| Train | 0.0018 | 0.0041 | 0.0026 | 0.0010 | 5.3890e-5 | 0.0013 | 8.1171e-4 |
| Test | 1.0230 | 0.1119 | 0.4155 | 0.7401 | 0.0283 | 0.2695 | 0.4006 |

According to the above tables, the lowest MMRE is related to TURN, VIRT and LOC parameters. So we focus on training the ANFIS based on these 3 input parameters. The results of the simulation are visible in the Figure (3) and (4).

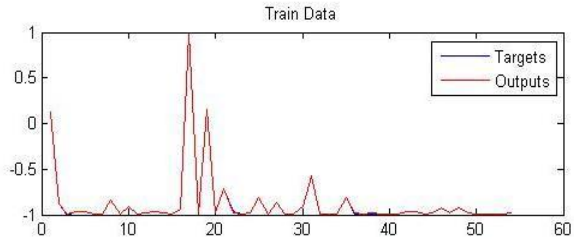


**Figure 3. Output Diagram of the Proposed Model and Target of Training Data**

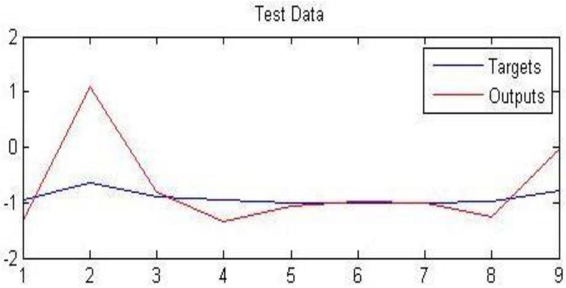


**Figure 4. Output Diagram of the Proposed Model and Target of Training Data**

Result of the proposed model based on the criteria of MMRE, RMSE and PRED (25) is shown in Table (5).

**Table 5. Results of Criteria in Proposed Model**

| Performance Criteria | MMRE | RMSE | PRED(25) |
|---|---|---|---|
| **Train** | 2.623e-4 | 0.0011 | 0.962 |
| **Test** | 0.0984 | 0.1539 | 0.889 |

## 5. Conclusion and Future Work

In this paper, we estimate the cost of software development by using adaptive neuro-fuzzy inference systems. The aim of this paper was to increase the accuracy of cost estimates for new software projects. Then, due to the limited number of ANFIS inputs, we used trial and error and finally we considered the parameters of TURN, VIRT and LOC as ANFIS inputs. Finally, we were able to provide the suitable architecture for estimating software projects cost based on ANFIS with MMRE error equal to 2.623e-4 for the training data and 0.0984 for testing data. Also, we estimated the obtained accuracy of the proposed ANFIS architecture using PRED (25). This accuracy for training data obtained to be 96% and for testing data is 89%, it is hoped that the proposed method will be very effective in estimating the cost development of new software.

## References

[1] R. Bhatnagar, V. Bhattacharjee, and M.K. Ghose, M.K, Software Development Effort Estimation Neural Network Vs, Regression Modeling Approach, International Journal of Engineering Science and Technology, Vol. 2, No.7, pp. 2950-2956, 2010.

[2] F.S.Gharehchopogh, A. Talebi, and I. Maleki, Analysis of Use Case Points Models for Software Cost Estimation, International Journal of Academic Research, Part A, Vol. 6, No.3, pp.118-124, 2014.

[3] F.S. Gharehchopogh, I. Maleki, and A. Talebi, Using Hybrid Model of Artificial Bee Colony and Genetic Algorithms in Software Cost Estimation, 9th International Conference on Application of Information and Communication Technologies (AICT), IEEE, Rostov on Don, pp. 102-106, 2015.

[4] A.S. Pressman, Software Engineering: a practitioner's Approach, New York: MCGrawHill, Edition 7, Pages.930, 2005.

[5] F.S. Gharehchopogh, L. Ebrahimi, I. Maleki, and S.J. Gourabi, A Novel PSO based Approach with Hybrid of Fuzzy C-Means and Learning Automata in Software Cost Estimation, Indian Journal of Science and Technology, Vol.7, No.6, pp.795-803, 2014.

[6] T. Menzies, Z. Chen, J. Hihn, and K. Lum, Selecting Best Practices for Effort Estimation, IEEE Transactions on Software Engineering, Vol.32, No.11, pp. 883-895, 2006.

[7] I. Maleki, L. Ebrahimi, and F.S. Gharehchopogh, A Hybrid Approach of Firefly and Genetic Algorithms in Software Cost Estimation, MAGNT Research Report, Vol.2, No.6, pp. 372-388, 2014.

[8] K.M. Mewada, A. Sinhal, and B. Verma, Adaptive Neuro-Fuzzy Inference System (ANFIS) Based Software Evaluation. IJCSI International Journal of Computer Science Issues, Vol.10, No.5, pp.244-250, 2013.

[9] S.P.Rama, S.P.V.G.D. Prasad, and K.R. Sudha, Hybrid Neuro-Fuzzy Systems for Software Development Effort Estimation. International Journal on Computer Science and Engineering, Vol.4, No.12, pp.1924-1932, 2012.

[10] E. Praynlin, and P. Latha, Estimating Development Effort of Software Projects using ANFIS, International Conference on Recent Trends in Computational Methods, Communication and Controls, Ongole, Andhra Pradash, India, pp.15-20, 2012.

[11] S. Dhingra and P.S. Mann, An Adaptive Neuro Fuzzy Approach for Software Development Time Estimation, International Journal of Advanced Research in Computer Science and Software Engineering, Vol.3, No.5, pp.586-590, 2013.

[12] K.S. Ziauddin, S., Khan, and J.A. Nasir, A Fuzzy Logic Based Software Cost Estimation Model, International Journal of Software

Engineering and Its Applications, Vol.7, No.2, pp.7-18, 2013.

[13] U.R. Saxena and S.P. Singh, Software Effort Estimation Using Neuro-Fuzzy Approach, 2012 CSI Sixth International Conference on Software Engineering, Devi Ahilya University Indore, Madhay Pradesh, India, pp.1-6, 5-7 Sep 2012.

[14] Z.A. Khalifelu and F.S. Gharehchopogh, Comparison and Evaluation Data Mining Techniques with Algorithmic Models in Software Cost Estimation. Elsevier Press, Procedia-Technology Journal, Vol.1, pp.65-71, 2012.

[15] B.T. Rao, B. Sameet, G.K. Swathi, K.V. Gupta, CH. RaviTeja, and S. Sumana, A Novel Neural Network Approach for Software Cost Estimation Using Functional Link Artificial Neural Network (FLANN), IJCSNS International Journal of Computer Science and Network Security, Vol.9, No.6, pp.126-131, 2009.

[16] M.K. Luka and A.A. Atayero, ANFIS Models for Dynamic Load Balancing in 3GPP LTE: A Fresh Take on Load Balancing in NGN, LAP Lambert Academic Publishing, Germany, 2012.

[17] PROMISE Software Engineering Repository, http://promise.site.uottawa.ca/SERepository/datasets/cocomo81.arff, [Last Available: 2016.02.04].

[18] T. Jayalakshmi and A. Santhakumaran, Statistical Normalization and Back Propagation for Classification. International Journal of Computer Theory and Engineering, IACSIT, Vol.13, No.1, pp.89-93, 2011.