To Design and Implement Neural Network and Fuzzy Logic for Software Development Effort Prediction

Shiyna Kumar
Student
DAV Institute of Engineering & Technology,
Kabir Nagar, Jalandhar
PUNJAB,

Vinay Chopra
Assistant Professor
DAV Institute of Engineering & Technology,
Kabir Nagar, Jalandhar
PUNJAB

ABSTRACT
One of the greatest challenges for software developers is forecasting the development effort for a software system for the last decades. The capability to provide a good estimation on software development efforts is necessitated by the project managers. Software effort estimation model divided into two main categories: algorithmic and non- algorithmic. These models too have difficulty in modeling the inherent complex relationships between the contributing factors, are unable to handle categorical data as well as lack of reasoning capabilities. The limitations of these models led to the exploration of the techniques which are soft computing based. In This paper we have compared neural network and fuzzy logic model for software development effort estimation. It will help us to make accurate software effort estimation by these estimation techniques

General Terms
Fuzzy logic, neural network, Software effort estimation

Keywords
Fuzzy Logic, Neural Network (FFNN, RBNN), Prediction, MRE, MMRE, BRE, Development Effort

1. INTRODUCTION
Software Engineering is the systematic approach to the development, maintenance and retirement of the software. There are various problems that software engineering faces. The main problem that occurs in software engineering is when we are not reliable to predict the effort required to build a system. Software effort estimation is a necessary feature that guides and supports the planning of software projects. Effort estimation is generally offers to estimate the effort required to develop the software. Software effort estimation refers to the predictions of the likely amount of effort, time, and staffing levels required to build a software system. An extremely helpful form of effort prediction is the one made at an early stage during a project, when the costing of the project is proposed for approval. Effort estimation algorithms [1] in general offer estimates of the number of work months required to produce a given amount of code. Age old approaches for software projects effort prediction such as the use of mathematical formulae derived from past data, or the use of expert's judgments, lack in terms of efficiency and robustness in their results. Software effort estimation guides the prediction of the likely amount of effort, time, and staffing levels required to build a software system at an early stage during a project. However, estimates at the preliminary stages of the project are the most difficult to obtain because the primary source to estimate the costing comes from the requirement specification documents[2].According to Royce[3], a good and effective software cost estimate should fulfill the different types of properties. One is conceptualized and supported by the software project manager and the development team and another is acknowledged by all the stake holders as achievable. The underlying cost model is well-defined on a credible basis. It is based on the careful analysis of the relevant historical project data (similar processes, similar technologies, similar environments, similar people and similar requirements).It is defined in sufficient detail such that its possible key risk areas are clearly understood and probability of success is objectively assessed. There are different approaches that you can use to estimate effort i.e. Bottom-Up estimation and Top-Down estimation. In this paper, we present a fuzzy logic for software development effort estimation. In this proposed method accurate effort estimation will be done by using fuzzy logic and neural network models and the results of fuzzy logic will be compared with RBNN based upon various parameters such as Magnitude of Relative Error (MRE), Mean Magnitude of Relative Error (MMRE), Balanced Relative Error (BRE) and Prediction (Pred).

2. RELATED WORK
Lopez Martin et al. [35, 36] proposed a fuzzy logic model for development effort estimation. Vachik S. Dave et al.[28] compared a popular techniques in software development effort estimation such as neural network models and regression models. Their result shows that neural network models shows better result than regression models. Sandeep kad et al. [37] observed that Gaussian membership function performs better results than triangular and trapezoidal membership function. This work employs a comparison between neural network approach and fuzzy logic model for software development effort estimation.

3. METHODOLOGY
The main goal of this paper is to evaluate software development effort using a fuzzy logic and RBNN. In this paper a neural network tool is used. The network is trained by using learning algorithm i.e. back propagation method. This methodology consists of following steps:

Step-1: Define the input variable membership function.

a. Define fuzzy sets for size input variable using Gaussian MF's.
b. Define fuzzy sets for mode input variable using Gaussian MF's.

Step-2: Defining the output variable functions as a nominal effort.

Step-3: Create a fuzzy inference system with a rule base formulated using the knowledge of relationships between mode, size and effort.

Step-4: Define a fuzzy inference system for each cost driver using a 6 point scale ranging from very low to extra high.

Step-5: Calculate EAF by multiplying the value of all the cost drivers

Step-6: Evaluate the final effort by integrating the two components i.e. The Nominal effort and EAF.

Step-7: Design a Feed-forward neural network and Radial Basis neural network with the 10 hidden layers.

Step-8: Train These Neural networks with 50 randomly selected projects and by taking the value of epoch 2500 and save these networks as FFNN and RBNN.

Step-9: Evaluate the value of Effort with these saved neural networks.

Step-10: Compare the FFNN and RBNN for software development effort estimation on the basis of MMRE and Prediction.

Step-11:-Evaluate and Compare the Value of MMRE, BRE and PRED from result obtained by best neural network and Fuzzy Inference System.

3.1 Fuzzy logic Model

According to the Oxford English Dictionary, the word Fuzzy is defined as blurred, indistinct, imprecisely defined, confused or vague. Fuzzy systems are knowledge based or rule based systems [18]. The heart of a fuzzy system is a knowledge base consisting of the so called fuzzy IF-THEN rules. A fuzzy IF-THEN rule is an IF-THEN statement in which some words are characterized by continuous membership functions. Thus fuzzy logic can be used to handle the imprecision and uncertainty present in the early stages of the project to predict the effort more accurately by incorporating total transparency in the prediction system. This proposed method estimates the software development effort accurately by taking the size and mode as a input. In our design, we use Gaussian membership function to represent the effort.

The below figure 1 shows the effort estimation using 2 Inputs that is Input 1 (Size) and Input 2 (Mode) and an Output (Effort).

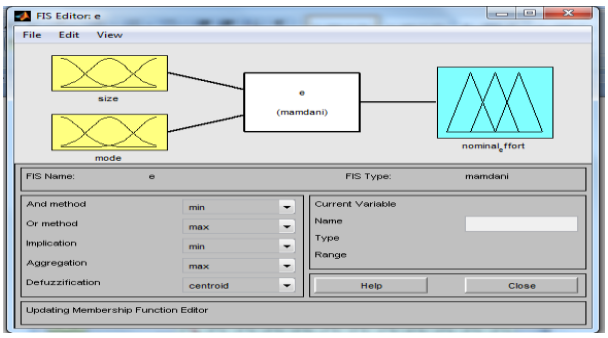

Fig 1: Fuzzy Set for Effort Estimation

3.2 Neural Network Modeling

The network contains neurons arranged in layers with each neuron is connected to every neuron of the lower layer forming a complete graph. The basic architecture consists of three types of neuron layers: input, hidden, and output layers. The data processing can extend over multiple (layers of) units, but no feedback connections are present. The feed forward and radial basis multi-layer networks with back propagation learning is the most commonly used structure in the field of software effort estimation. In our experiment, for training and testing of prediction models, we divided dataset into two parts. Training set contains 50 projects of the dataset. 7 projects are used for testing of the models. This particular experiment is conducted to compare the accuracy level of prediction capability of neural network with respect to previous trained neural network. FFNN and RBNN are trained by assuming learning rate as 0.75 and the training iterations used is 2500. Using this training configuration we trained the FFNN and RBNN by taking input as a 15 cost drivers and LOC of each projects. The target value will be set as actual effort for each project. After that we find the estimated values for test dataset using trained network. After evaluating the value of efforts with these neural networks, we have compared the RBNN and FFNN

FFNN and RBNN are conducted to compare the accuracy level of prediction of neural network. For our experiment we are using simple FFNN and RBNN with three layers: Input layer, Hidden layer and Output layer.

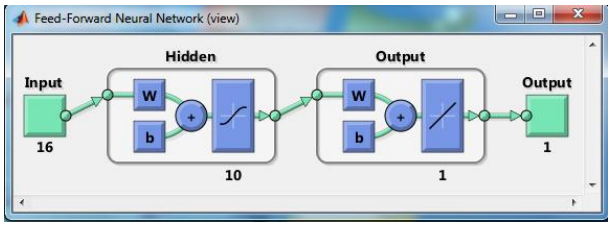

Fig 2: Neural network view for feed forward neural network

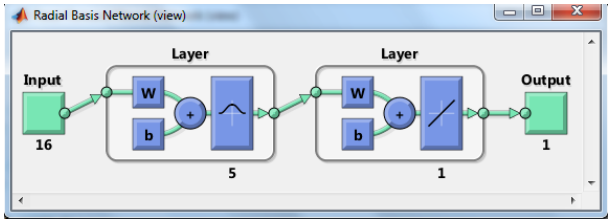

Fig 3: Neural network view for radial basis neural network

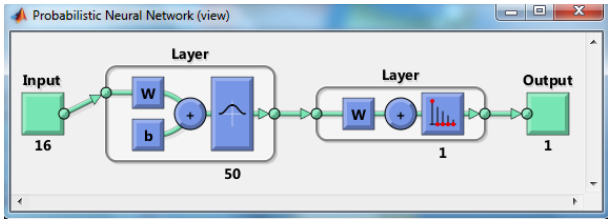

Fig 4: Neural network view after training the data.

Table 1. Comparison between FFNN and RBNN

Performance Criteria	FFNN	RBNN
MMRE (%)	70.6870	37.3027
Pred (25%)	42.86	42.86
BRE	0.8312	0.6991

We have evaluated these prediction models with three evaluation criteria: Mean Magnitude Relative Error (MMRE), Prediction (n) and BRE. A model which provides higher value of Pred (n) is better than one with lower value of Pred (n); while model with lower MMRE is better than one with higher MMRE and the model having lower BRE is better than one with higher BRE. So in this experiment RBNN shows better results.

3.3 Performance Evaluation Metrics

The following evaluation metrics are adapted to assess and evaluate the performance of the effort estimation models.

 Magnitude of Relative Error (MRE) is a common criterion for the evaluation of effort estimation model is the magnitude of relative error (MRE). MRE is defined as follow:

$$MRE = \frac{|Actual\ Effort - Estimated\ Effort|}{Actual\ Effort} \times 100 \qquad \text{Eq. (1)}$$

 Mean Magnitude of Relative Error (MMRE)

The MMRE calculates the mean for the sum of the MRE of n projects. Specifically, it is used to evaluate the prediction performance of an estimation model. It basically computes the average of MRE over n projects.

$$MMRE = \frac{1}{n}\sum_{i=1}^{i=n}\frac{|\text{Actual Effort} - \text{Predicted Effort}|}{\text{Actual Effort}} \qquad \text{Eq. (2)}$$

 Prediction Level (PRED)

Prediction at level n is defined as the % of projects that have absolute relative error less than n. Another widely used prediction quality indicator is pred(l), which is simply the percentage of estimates that are within of the actual value Typically m is set to 25.

$$PRED\ (l) = \frac{k}{n} \times 100 \qquad \text{Eq. (3)}$$

Where l is the maximum MRE of a selected range, n is the total number of projects, and k is **number of projects** in a set of n projects whose MRE <= l.

 Balanced Relative Error (BRE) this is another parameter to evaluate the effort estimation. Where BRE is define as follows:

$$BRE(\%) = \frac{|Estimated\ Effort - Actual\ Effort|}{\min(T, T')} \times 100 \qquad \text{Eq. (4)}$$

Where T= estimated Effort and T'=actual Effort

3.4 Proposed Framework

This proposed framework developed an optimized fuzzy logic based framework and reconstructs the Neural Network model for software effort estimation. To evaluate development effort we have used COCOMO NASA data set on proposed developed models. This research is used to handle the imprecision and uncertainty present in the early stages of the project to predict the effort more accurately by incorporating total transparency in the prediction system. This model computes effort as a function of program size and a lot of cost drivers that includes subjective assessment of product attributes, hardware attributes and project attributes. To estimate the effort more accurately we have compared the neural network model and fuzzy logic framework. The evaluation of the models is based upon the following parameters i.e. Magnitude of Relative Error (MRE)**,** Mean Magnitude of Relative Error (MMRE), Prediction Accuracy (Pred) and Balanced Relative Error (BRE).

4. Results and Discussion

For experimental analysis, we have chosen 7random projects from COCCOMONASA data set. It shows that the proposed model has MMRE less than RBNN model as shown in Figure 4. The comparisons between the results are shown in table 3.

The Below graph shows that fuzzy logic having lower MMRE then RBNN and also fuzzy logic model Shows Higher prediction value then RBNN, Thus, after comparison between fuzzy logic model and RBNN model based upon two parameters, Fuzzy Logic Model shows better results.

Table 2. Comparison between FFNN and RBNN

Projects	LOC	Actual Effort	Fuzzy logic effort	Fuzzy logic MRE	RBNN effort	RBNN MRE
1	423	2300	510.0000	.77	2300	0.00
2	79	400	406.1592	.01	176.8	0.5580
3	284.7	973	776.1119	.20	1224.1	0.2581
4	282.1	1368	782.6257	.42	1129.5	0.1743
5	78	571.4	401.5486	.29	176.8	0.6906
6	11.4	98.8	114.0779	.15	176.8	0.7896
7	19.3	155	141.2874	.08	176.8	0.1407

.

Table 3. Comparison between Fuzzy logic and RBNN

Performance Criteria	Fuzzy Logic	RBNN
MMRE (%)	28.052	37.3027
Pred (25%)	57.14	42.86
BRE	0.7430	0.6991

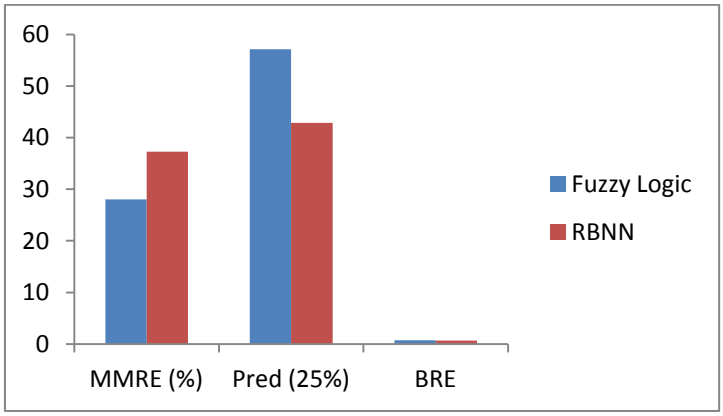

Fig 5: Comparison of different results.

5. CONCLUSIONS

This research is used to handle the imprecision and uncertainty present in the early stages of the project to predict the effort more accurately by incorporating total transparency in the prediction system. A model which gives lower MMRE is better than that which gives higher MMRE. A model which gives higher Pred(n) is better than that which gives lower Pred (n). Hence from the Table 3 we can observe that, based upon two parameters Fuzzy Logic Model is better than Radial basis neural network.

6. ACKNOWLEDGEMENT

The authors are greatly indebted to the Department of Computer Science and Engineering, D.A.V Institute of Engineering and Technology, Jalandhar for providing excellent lab facilities that make this work possible.

7. REFERENCES

[1] Moataz A. Ahmed, Moshood Omolade Saliub, Jarallah AlGhamdia "Adaptive fuzzy logic-based framework for software development effort prediction", Information and Software Technology, pp. 31–48, 2005.

[2] Harsh Kumar Verma, Vishal Sharma "Handling Imprecision in Inputs using Fuzzy Logic to Predict Effort in Software Development" , IEEE conference, 2010.

[3] Ch. Satyananda Reddy, KVSVN Raju " An Improved Fuzzy Approach for COCOMO's Effort Estimation using Gaussian Membership Function" Journal Of Software, Vol. 4, No. 5, July 2009.

[4] Iman Attarzadeh, Siew Hock Ow "Improving Estimation Accuracy of the COCOMO II Using an Adaptive Fuzzy Logic Model", IEEE International Conference on Fuzzy Systems June 27-30, 2011, Taipei, Taiwan.

[5] G. D. Boetticher, "An assessment of metric contribution in the construction of a neural network- based effort estimator", Proceedings of Second International Workshop on Soft Computing Applied to Software Engineering, 2001.

[6] A.R. Venkatachalam, "Software cost estimation using artificial neural networks", in: Proceedings of the International Joint Conference on Neural Networks, pp. 987–990, 1993.

[7] M.O. Saliu, M. Ahmed, "Soft computing based effort prediction systems—A survey", in: E. Damiani, L.C. Jain (Eds.), Computational Intelligence in Software Engineering, Springer-Verlag, ISBN 3-540-22030-5, July 2004.

[8] Moataz A. Ahmed, Moshood Omolade Saliu and Jarallah AlGhamdi, "Adaptive fuzzy logic-based framework for software development effort prediction", Information and Software Technology, 2005.

[9] Anish Mittal,Kamal Parkash,Harish Mittal " Software Cost Estimation Using Fuzzy Logic", ACM SIGSOFT Software Engineering Notes Page 1, Volume 35 No., November 2010.

[10] Mrinal Kanti Ghose, Roheet Bhatnagar and Vandana Bhattacharjee. "Comparing Some Neural Network Models for Software Development Effort Prediction", IEEE Conference, 2011.

[11] J.S. Pahariya, V. Ravi and M. Carr, "Software Cost Estimation using Computational Intelligence Techniques", World Congress on Nature & Biologically Inspired Computing(NaBIC 2009) IEEE, 2009.

[12] C. C. Kung and J. Y. Su, "Affine Takagi-Sugeno fuzzy modeling algorithm by Fuzzy c-regression models clustering with a novel cluster validity criterion", IET Control Theory Appl., pp. 1255 – 1265, 2007.

[13] T. Gruschke, "Empirical Studies of Software Cost Estimation: Training of Effort Estimation Uncertainty Assessment Skills", 11th IEEE International Software Metrics Symposium, IEEE, 2005.

[14] S Basha and P. Dhavachelvan, "Analysis of Empirical Software Effort Estimation Models", International Journal of Computer Science and Information Security(IJCSIS), Vol. 7, No. 3, 2010.

[15] B. W. Boehm, "Software Engineering Economics", Prentice-Hall, Englewood Cliffs, NJ, USA, 1981.

[16] B.Hughes and M.Cotterell, "Software Project Management", Tata McGraw-Hill, 2006.

[17] S.G. MacDonell, A.R. Gray, "A comparison of modeling techniques for software development effort prediction", in: Proceedings of the International Conference on Neural Information Processing and Intelligent Information Systems, Dunedin, New Zealand, Springer, Berlin, pp. 869–872.1997.

[18] Zonglain, F. and xihui L.,"F-COCOMO:Fuzzy Construtive Cost Model In Software Engineering" , In Proc. Of IEEE International Conference On Fuzzy Systems., pp.331-337,1992.

[19] J. Keung, "Software Development Cost Estimation Using Analogy: A Review", Australian Software Engineering conference, IEEE, 2009.

[20] Vachik S. Dave, Kamlesh Dutta, "Neural Network based Software Effort Estimation & Evaluation criterion MMRE", International Conference on Computer & Communication Technology (ICCCT), pp.347-351, 2011.

[21] Heejum Park and Seung Baek, "An Empirical validation of a neural network model for software effort estimation", Expert System with Application, Elsevier, pp.929-937, 2007.

[22] J. Ryder, "Fuzzy modeling of software effort prediction", Proceedings of IEEE Information Technology Conference, Syracuse,NY, 1998.

[23] Cuauhtémoc López Martín, "Software Development Effort Estimation Using Fuzzy Logic: A Case Study", Proceedings of the Sixth Mexican International Conference on Computer Science (ENC'05), IEEE 2005.

[24] N. Sharma1, A. Bajpai and M. R. Litoriya, "A Comparison of Software Cost Estimation Methods: A Survey", The International Journal of Computer Science and Applications (TIJCSA), Vol.1, no. 3, ISSN – 2278 – 1080, May 2012.

[25] V. Khatibi, Dayang and N. A. Jawawi, " Software Cost Estimation Methods: A Review", Journal of Emerging Trends in Computing and Information Sciences, CIS Journal, Vol. 2, no. 1, ISSN 2079-8407, 2011.

[26] Jaswinder Kaur, Satwinder Singh, Dr. Karanjeet Singh Kahlon, Pourush Bassi, " Neural Network-A Novel Technique for Software Effort Estimation", International Journal of Computer Theory and Engineering, Vol. 2, No.1, pp.1793-8201,Feb. 2010.

[27] Alaa Sheta, David Rine and Aladdin Ayesh "Development of Software Effort and Schedule Estimation Models Using Soft Computing Techniques", IEEE Congress on Evolutionary Computation, pp.1283-1289, 2008.

[28] Vachik S. Dave, Kamlesh Dutta, "Comparison of Regression model, Feed-forward Neural Network and Radial Basis Neural Network for Software Development Effort Estimation", ACM SIGSOFT Software Engineering, Vol. 36, No.5, Sept. 2011.

[29] A. García, I. González, R. Colomo, J.L. López and B. Ruiz, "Methodology for Software Development

Estimation Optimization Based on Neural Networks", Latin America transactions, IEEE, Vol.9, No.3, pp. 384-398, 2011.

[30] Nonika Bajaj, Alok Tyagi and Rakesh Agarwal, "Software Estimation –A Fuzzy Approach", ACM SIGSOFT Software Engineering, Vol. 31, No. 3, May 2006.

[31] Mrinal Kanti Ghose, Roheet Bhatnagar and Vandana Bhattacharjee, "Software Development Effort Estimatiom- Neural Network Vs. Regression Modelling Approach", International Journal of Engineering and Technology, Vol. 2(7), pp. 2950-2956,2010.

[32] S.G. MacDonell,"Software source code sizing using fuzzy logic modeling," Information and Software Technology, pp.389–404, 2003.

[33] C. Schofield,"Non-algorithmic effort estimation techniques, Technical Reports," Department of Computing, Bournemouth University, England, TR98-01, March 1998.

[34] Xishi Huang, Luiz F. Capretz, Jing Ren, Danny Ho "A Neuro-Fuzzy Model for Software Cost Estimation" Proceedings of the Third International Conference On Quality Software (QSIC'03) IEEE, 2003.

[35] C. Lopez-Martin, C.Yanez-Marquez, A.Gutierrez-Tornes, "Predictive accuracy comparison of fuzzy models for software development effort of small programs", The journal of systems and software, Vol. 81, Issue 6, 2008, pp. 949-960.

[36] C.L. Martin, J.L. Pasquier, M.C. Yanez, T.A. Gutierrez, "Software Development Effort Estimation Using Fuzzy Logic: A Case Study", IEEE Proceedings of the Sixth Mexican International Conference on Computer Science (ENC'05), 2005, pp. 113.

[37] Sandeep Kad, Vinay Chopra, "Software Development Effort Estimation Using Soft Computing", International journal of Machine Learning and Computing, Vol 2, No.5, pp.548-551, October 2012.