Volume 3, Issue 5, May 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

An Adaptive Neuro Fuzzy Approach for Software Development Time Estimation

Shina Dhingra, Palvinder Singh Mann
Department of Computer Science and Engineering, D.A.V.I.E.T.
Jalandhar, Punjab, India.

***Abstract:** This paper presents an Adaptive Neuro-Fuzzy Approach for Software Development Time Estimation. This proposed technique is aimed at building and evaluating a Neuro - fuzzy model for software project development time. . The forty one modules were used as a data set. Our proposed approach is compared with neural network models and the results show that values of various relative error parameters for Neuro-fuzzy is lower than the values of parameters applying neural network.*

***Keywords:** Adaptive, Neuro-Fuzzy, Inference System (ANFIS), Neural Network, Fuzzy Logic, MRE. MMRE, BRE, Prediction.*

1. INTRODUCTION

To develop a project successfully, it is essential for any organization that the project should be completed within budget, on time and the project should have required quality. Many existing research papers have proposed various estimation techniques, but no single software development estimation technique is the best for all situations [24].The neural network research started in the 1940s, and the fuzzy logic research started in the 1960s, but the Neuro-fuzzy research area is relatively new [32]. The objective of this paper is to present a possible way of combining fuzzy logic and neural networks for achieving higher accuracy. Once the concept of fuzzy logic is incorporated into neural network, the result is a Neuro-fuzzy system that combines the advantages of both techniques [33]. A software tool (MATLAB 7.10) was used to process the Neuro-fuzzy systems. Accurate time estimation is a crucial skill in project management. Without it, you won't know how long your project will take, and you won't be able to get commitment from the people who need to sign it off. Even more importantly for your career, sponsors often judge whether a project has succeeded or failed depending on whether it has been delivered on time and on budget. To have a chance of being successful as a project manager, you need to be able to negotiate sensible budgets and achievable deadlines. Once you've estimated the time needed for each task, you can prepare your project schedule. Add your estimates to the draft activity list that you produced in the second step, above. You can then create a Gantt chart to schedule activities and assign resources to your project; and to finalize milestones and deadlines. There are different approaches that you can use to estimate time. Bottom-Up estimating, Top-Down estimating, comparative estimating, parametric estimating and three Point estimating. There are various steps to estimate time accurately. Understand What's Required, Order these activities, Decide who you need to involve, Make your estimates, Think of some Buffer Time, Availability of all the resources for Estimated Period, Think of your past Experience to make judgment. In the system described in [26], they have proposed different types of Neuro- Fuzzy Models. Two case studies have been used for this purpose. The first is based on NASA-93 dataset and the other is based on Maxwell-62 dataset. The suitable Neuro- Fuzzy Model for a dataset depends on two factors: Linearity in the dataset and No. of Fuzzy Rules. The case studies were analyzed using six different criteria. In the system described in [27], they have explored Neuro-Fuzzy techniques to design a suitable model to utilize improved estimation of Software effort for NASA software projects. Comparative analysis between Neuro-Fuzzy model and the traditional software model has been done. Integration of neural networks, fuzzy logic and algorithmic models into one scheme has resulted in providing robustness to imprecise and uncertain inputs. In this paper, we present an adaptive Neuro fuzzy inference system (ANFIS) for software development time estimation. In the proposed method accurate estimation of software development time will be done and the results of Neuro Fuzzy approach will be compared with different types of neural network models 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. ANFIS METHOD

The proposed method estimates the software development time accurately by proposed Adaptive Neuro Fuzzy Inference System (ANFIS) as it is a combination of Fuzzy Logic and Neural Network so ANFIS takes advantages from fuzzy logic and

neural network. This ANFIS constructs a Fuzzy inference system by using given training data set whose membership function parameters are adjusted by back propagation algorithm or in combination with least square type of method. Figure.1 shows a high level diagram of the proposed ANFIS. Inputs and their membership functions appear to the left of the ANFIS structural characteristics, while outputs and their membership functions appear on the right.

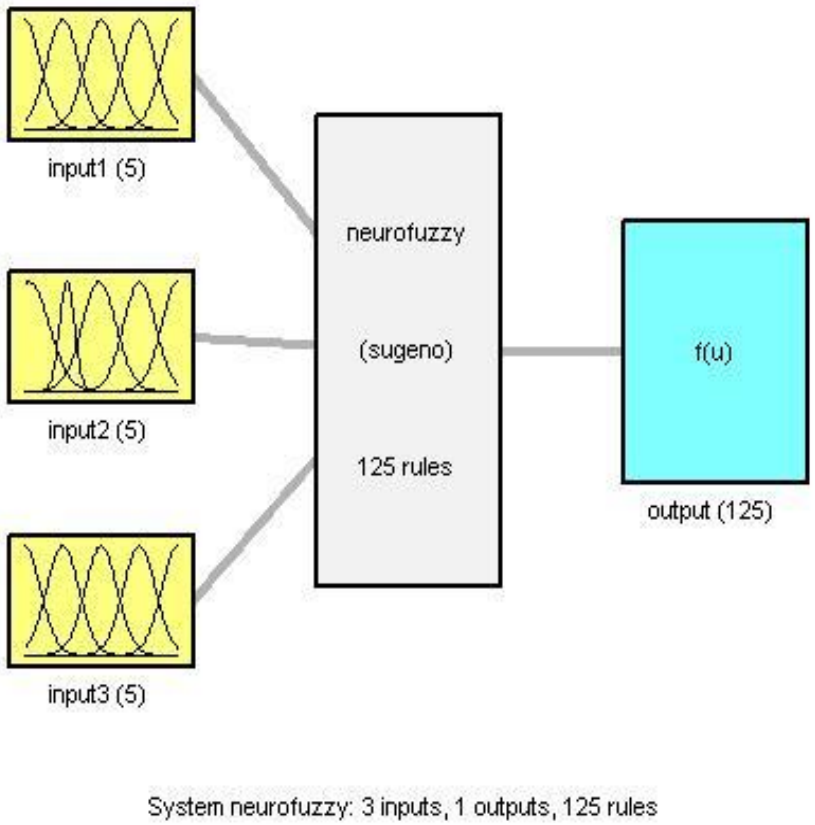

Figure 1: Diagram of proposed ANFIS

2.1 PROPOSED METHOD

The proposed ANFIS model is a first-order Sugeno type fuzzy inference system with 3-inputs and 1-output. Each input has generalized bell type membership functions and the output has a constant membership function.

To estimate the time accurately, first we select the modules from the Lopez Martin data set which have least variations and then train our ANFIS model with the training data as shown in Table 1.

Table1: Training Data

INPUTS			OUTPUT
MC	DC	LOC	
2	0.083	10	15
2	0.125	9	15
2	0.125	9	16
2	0.125	14	16
2	0.167	7	16
2	0.167	8	18
2	0.167	10	15
2	0.167	10	15
2	0.167	10	18
2	0.2	10	14
2	0.2	10	15

3	0.083	17	22
3	0.125	11	19
3	0.125	15	18
3	0.125	15	19
3	0.143	13	21
3	0.143	14	20
3	0.143	14	21
3	0.143	15	19
3	0.143	15	20
3	0.167	13	15
3	0.167	14	13
3	0.2	18	19
5	0.143	22	24.5
5	0.143	22	24.5
4	0.077	16	21
4	0.077	31	21

Therefore, the number of rows of the training data is equal to the number of training data pairs, and, since there is only one output, the number of columns training data is equal to the number of inputs plus one.

3. RESULTS

We have divided the entire dataset into two sets, training set and validation set in the ratio of 100% and 20%. Training set consists of data from 30 projects selected randomly and validation set consists of 12 project samples that were used later on for testing the generalization ability of the best trained model, i.e., the ANN that yielded the lowest MMRE. We randomly initialized the weights and momentum coefficients and re-trained the network 10 times with the back propagation algorithm. Finally, we utilized the best ANN to proceed to the testing phase.

Table1: The DT obtained by different Models

Project no	Actual DT	Estimated DT' using FFBP NN	Estimated DT' BY using CASCADED FFBP NN	Estimated DT' BY using layer RECURRENT NN	Estimated DT' using Neuro-fuzzy
31	19	9.0048	9..9942	9.0726	19
32	13	9.1982	9.0578	10.8072	12.7
33	12	9.0653	9.0156	9.1280	12.4
34	12	9.0100	9.0096	9.0001	11.9
35	21	21.9776	20.5233	10.0386	21
36	21	21.9869	21.8353	9.0006	21
37	19	21.8789	20.0193	9.1283	19
38	18	9.0251	17.7382	9.000	18
39	24	9.0392	15.985	9.3082	24.5
40	25	9.0392	15.98	9.3082	24.5
41	18	9.2389	17.11	9.0000	18

Table1: THE MMRE AND PRED COMPARISON BETWEEN ESTIMATION MODELS

Performance Criteria	FFBP NN	CASCADED FFBP NN	RECURRENT NN	Neuro-fuzzy
MMRE (%)	34.66	19.25	45.7	6.6
Pred (25 %)	27.27	54.54	27.2	90.9

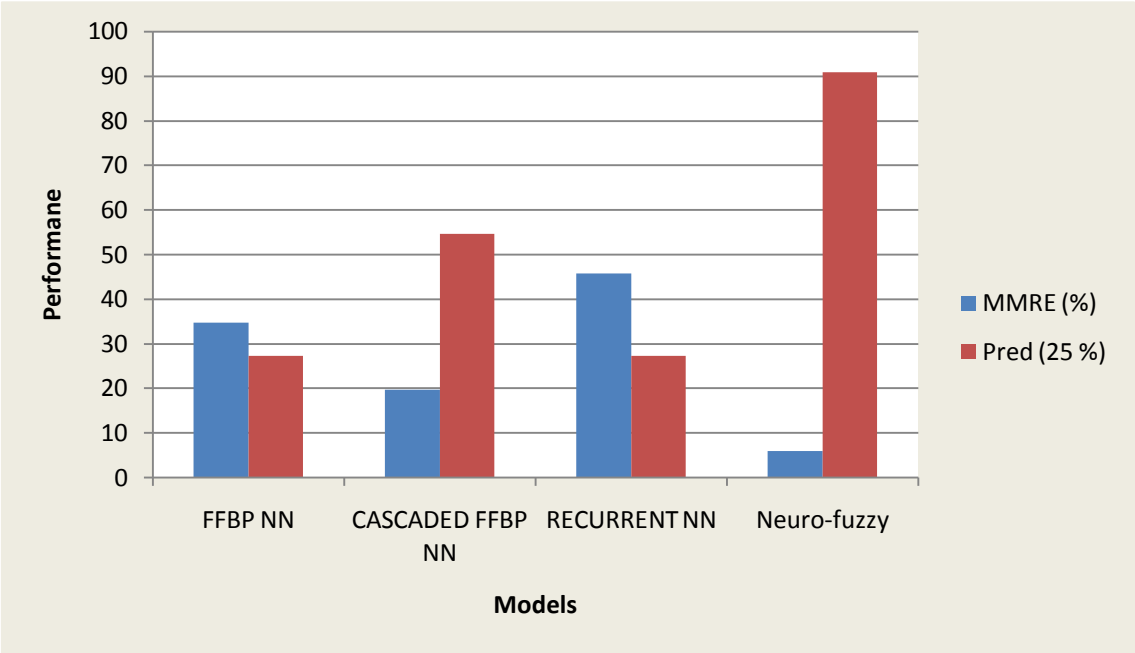

Figure 2: Comparison of Different models

4. CONCLUSION

Software Development Time Estimation results show that the proposed ANFIS model gives outstanding results as compared to different types of neural network models. This model is suitable for various areas of software development such as Effort estimation, Cost Estimation.

5. 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.

REFERENCES

[1] S.Basha and P.Dhavachelvan, "Analysis of Empirical Software Effort Estimation Models" in International Journal of Computer Science and Information Security (IJCSIS), Vol. 7, 2010.
[2] B.Hughes and M.Cotterell, "Software Project Management", Tata McGraw-Hill, 2006.
[3] T.Gruschke, "Empirical Studies of Software Cost Estimation: Training of EffortEstimation Uncertainty Assessment Skills", 11th IEEE International Software Metrics Symposium, IEEE, 2005.
[4] B. W. Boehm, "Software Engineering Economics", Prentice-Hall, Englewood Cliffs, NJ, USA, 1981.
[5] C. C. Kung and J. Y. Su, "Affine Takagi-Sugeno fuzzy modeling algorithm byFuzzy c-regression models clustering with a novel cluster validity criterion", IETControl Theory Appl., pp.1255 – 1265, 2007.
[6] 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.
[7] 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.
[8] J. Keung, "Software Development Cost Estimation Using Analogy: A Review", Australian Software Engineering conference, IEEE, 2009.
[9] T. R. Benala, S. Dehuri and R. Mall, "Computational Intelligence in Software Cost Estimation: An Emerging Paradigm", ACM SIGSOFT Software Engineering Notes Page, Vol. 37, no.3, 2012.
[10] 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.
[11] Mrinal Kanti Ghose, Roheet Bhatnagar and Vandana Bhattacharjee, "Comparing Some Neural gNetwork Models for Software Development Effort Prediction", IEEE conference , 2011.
[12] Venus Marza, Amin Seyyedi and Luiz Fernando Capretz, "Estimating Development Time of Software Projects Using a Neuro Fuzzy Approach", World Academy of Science, Engineering and Technology, 2008.

[13] 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.

[14] 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.

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

[16] C.J. Burgess, M. Lefley, " Can genetic programming improve software effort estimation? A comparative evaluation", Information and Software Technology, pp.863–873, 2001.

[17] Anish Mittal, Kamal Parkash and Harish Mittal "Software Cost Estimation Using Fuzzy Logic", ACM SIGSOFT Software Engineering Notes Page , Vol 35 no. 1, November 2010.

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

[19] 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.

[20] Xishi Huang et al., "A Neuro-Fuzzy Model for Software Cost Estimation", Proceedings of the Third International Conference on Quality Software (QSIC'03), IEEE, 2003.

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

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

[23] Allidri et al., "COCOMO Cost Model Using Fuzzy Logic", 7th International Conference On RFuzzy Theory & Technology Atlantic City, New Jersey, February27-March3, 2000.

[24] 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.

[25] Dinesh C.S. Bisht and Ashok Jangid, "Discharge Modelling using Adaptive Neuro-Fuzzy Inference System", International Journal of Advanced Science and Technology,Vol. 31,pp.99-114, June 2011.

[26] Rama Sree P,Prasad Reddy and Sudha K.R., "Hybrid Neuro-Fuzzy Systems for Software Development Effort Estimation", International Journal on Computer Science and Engineering, Vol. 4, pp.1924-1932, Dec 2012.

[27] Urvashi Rahul Saxena, S.P.Singh, "Software Effort Estimation Using Neuro-Fuzzy Approach", CSI 6th International Conference, pp.1-6, Sept.2012.

[28] 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.

[29] Divya Kashyap, Ashish Tripathi,Prof. A.K.Mishra, "Software Development Effort and Cost Estimation: Neuro-Fuzzy Model", IOSR Journal of Computer Engineering (IOSRJCE), Vol. 2, pp. 12-14, July-Aug. 2012.

[30] M. Chemuturi, "Software Estimation Best Practices, Tools & Techniques: A Complete Guide for Software Project Estimator" , 2009.

[31] 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.

[32] J. Jantzen, "Neuro- Fuzzy modeling", Reort no 98-H-874, 1998.

