Software Effort Estimation Using Multilayer Perceptron and Adaptive Neuro Fuzzy Inference System

Berna Seref and Necaattin Barisci

***Abstract*—Accurate software effort estimation has a big importance for software companies for the reason that management of the project, control of the project, financial matters and timely deliveries are achieved with effort estimation. Thus, effort estimation plays vital role for software companies. In this study, software effort estimation is predicted by using Multilayer Perceptron and Adaptive Neuro Fuzzy Inference System. As a dataset, NASA 93 with 93 projects and Desharnais with 77 projects are used. The results show that Mean Magnitude Relative Error of Adaptive Neuro Fuzzy Inference System is lower than Multilayer Perceptron. In addition, it is seen that PRED(0.25) value of Adaptive Neuro Fuzzy Inference System is higher than Multilayer Perceptron. Thus, performance of Adaptive Neuro Fuzzy Inference System is better when compared to performance of Multilayer Perceptron.**

***Index Terms*—Software effort estimation, multilayer perceptron, adaptive neuro fuzzy inference system.**

I. Introduction

Software effort estimation has a big importance for software organizations and customers. Overestimation and underestimation can cause some vital problems not only for organizations but also customers. For example, if estimation is too low, the project must be completed as soon as possible. This situation can create negative impact on software developers and cause some faults. On the other hand, overestimation requires more resources in fact which is not needed and this situation causes budget fail [1]. Both overestimation and underestimation affect software organizations and customers in a negative way.

In order to predict software estimation, various software effort estimation methods are used. Algorithmic effort estimation, theory based techniques, empirical techniques, regression techniques and machine learning are one of these methods. It is not possible to say that a technique predicts the most accurate for all datasets and all situations. Some analysis of various machine learning methods must be done and they must be compared to get accurate effort estimation [2].

During the last years, a lof of researches are done with the aim of predicting and improving software effort estimation. For example, in a study, Takagi-Sugeno and Mamdami fuzzy models were used to predict software effort estimation. As a result, it was observed that Takagi-Sugeno shows beter performance when using projects with effort bigger than 100 [3]. In other study, effort estimation was done by using fuzzy model and statistical regression model. At the end of the experiment, it was observed that Mean Magnitude of Error Relative (MMER) of fuzzy model is lower or equal than regression model. In this study, when Analysis of Variance (ANOVA) was used in order to compare accuracies of the models, it was observed that there is no statistically significant difference between mean of fuzzy model and statistical regression model [4].

In this study, Multilayer Perceptron (MLP) and Adaptive Neuro Fuzzy Inference System (ANFIS) are used in order to predict software effort estimation. As a dataset, NASA93 with 93 projects and Desharnais with 77 projects are used. Mean Magnitude Relative Error (MMRE) and PRED (0.25) are used with the aim of comparing accuracies of them.

The rest of the paper is structured as follows. Section II explains the research background i.e. gives information about Multilayer Perceptron and Adaptive Neuro Fuzzy Inference System, describes the datasets used for prediction and explains various performance evaluation measures. Section III explains implimentation of this study. Results and discussions are given in Section IV. Finally, in the last section conclusions of the study is summarized and future works are explained.

II. Research Background

A. Multilayer Perceptron (MLP)

Multilayer Perceptron is a type of Neural Network. It consists of input layer, hidden layer and output layer. There is no computation in the input layer and it is responsible for receiving input vector and passing it the to the network. Hidden layers are used with the aim of improving performance of the network and a network may have one or more hidden layers. Number of neurons in the hidden layer must be chosen carefully for the reason that small number of neurons causes backpropagation algorithm not to be able to converge minimum point while training the system. On the other hand, more neurons cause overfitting and as a result, bad prediction performance [5]. Output layer represents solution that produced by network.

There are several neurons in each layers and these neurons are fully connected to all neurons in the previous and next layers. Each neuron calculates weighted sum of its inputs and adds bias term to this sum. After that passes the result to transfer function and applies this function. Result that

Manuscript received July 20, 2014; revised September 28, 2014.

B. Seref is with Faculty of Engineering, Department of Computer Engineering, Dumlupinar University, Kutahya, 43100, Turkey (e-mail: berna.seref@dpu.edu.tr).

N. Barisci is with the Faculty of Technology, Department of Computer Engineering, Gazi University, Ankara, 06500, Turkey (e-mail: nbarisci@gazi.edu.tr).

DOI: 10.7763/IJIMT.2014.V5.543

generated by this transfer function is output of this neuron and input for neurons which are in next layer [6].

B. Adaptive Neuro Fuzzy Inference System (ANFIS)

Adaptive Neuro Fuzzy Inference System is a type of artificial neural network and introduced by Jang in 1993 [7]. It is the combination of neural network and fuzzy logic. As a result, it can be said that disadvantages and limitations of these methods are reduced [8].

The general structure of an ANFIS model is given in Fig. 1 [9].

As it is seen in the Fig. 1, ANFIS consists of five layers. In this structure, x and y are input nodes, *A* and its derivations are linguistic labels. The outputs of nodes in the layer 1 are membership values. In layer 2, each node output is firing strength of a rule that calculated by mathematical multiplication. In layer 3, the normalization of the firing strengths is done. In layer 4, each output is a product of inputs that multiplied by the normalized firing strengths. In layer 5, final output which is summation of layer 4's outputs is calculated [10].

C. Data Collection

In this study, NASA 93 [11] and Desharnais [11] datasets are used in order to train and test the system.

NASA 93 dataset consists of 93 NASA projects. These projects are avionics, application ground, avionics monitoring, batch data processing, communications, data capture, launch processing, mission planning, monitor control, operating system, real data processing, science, simulation and utility projects that are collected from different NASA centers.

77 sample projects in Desharnais dataset are used in this study.

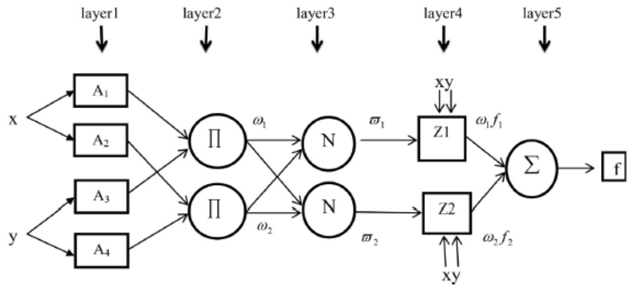

Fig. 1. Adaptive neuro fuzzy inference system's structure [9].

D. Performance Measures

There are several performance measures that are used with the aim of evaluating estimate capability of a model. Mean Magnitude of Relative Error (MMRE), PRED, Root Mean Squared Error (RMSE), and Mean Magnitude of Error Relative (MMER) are one of them. In this study, as an accuracy criterion, MMRE and PRED(0.25) are used.

Magnitude of Relative Error (MRE) is defined as follows:

$$\mathrm{MRE_i} = \frac{|\,\text{Actual Effort}_\mathrm{i} - \text{Predicted Effort}_\mathrm{i}\,|}{\text{Actual Effort}_\mathrm{i}} \tag{1}$$

In Equation (1), *i* is the number of the sample project.

Mean Magnitude of Relative Error (MMRE) is calculated as folllows:

$$\mathrm{MMRE} = \frac{1}{N}\sum_{i=1}^{N}\mathrm{MRE}_i \tag{2}$$

In Equation (2), while *N* represents total number of sample projects, *i* represents number of the sample project.

PRED(A) is used as a complementary criterion. The common value used for *A* is 25%. PRED(*A*) is defined as follows:

$$\mathrm{PRED}(A) = d\,/\,N \tag{3}$$

In equation (3), *N* is total number of projects, and *d* is number of projects with an error (MRE) less than or equal to *d*.

Root Mean Squared Error (RMSE) is defined as:

$$E = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n}(P_i - A_i)^2} \tag{4}$$

In equation (4), P_i is predicted value, A_i is actual value, *n* is total number of projects.

Mean Magnitude of Error Relative (MMER) is calculated by using Magnitude of Error Relative (MER) as follows:

$$\mathrm{MER}_i = \frac{|\text{Actual Effort}_i - \text{Estimated Effort}_i|}{\text{Estimated Effort}_i} \tag{5}$$

$$\mathrm{MMRE} = (1/n)\sum\nolimits_{i=1}^{n}\mathrm{MER}_i \tag{6}$$

In equation (5), *i* is project number. In equation (6), *n* is total number of projects.

III. Implementation

In this study, NASA 93 dataset consisting of 93 NASA projects and Desharnais dataset consisting of 77 projects are used. For ANFIS, the size by dataset used for training the system is 83 (NASA) and 66 (Desharnais), whereas the size by dataset to test the system is 10 (NASA) and 11 (Desharnais). For MLP, the size by dataset used for training the system is 75 (NASA) and 60 (Desharnais), whereas the total size by dataset to test and validate the system is 8 (NASA) and 6 (Desharnais).

As a performance evaluation criterion MMRE and PRED (0.25) are used.

Software effort estimations for 10 and 11 projects are predicted for both NASA 93 dataset and Desharnais dataset. Prediction of software effort estimation is based on multilayer perceptron and neuro fuzzy inferences system back propagation algorithm. Inputs are carried on Neural Fitting Tool (NFTool) and Neuro Fuzzy Inference System (ANFIS) Tool on matlab platform.

Firstly, inputs are carried on Multilayer Perceptron (MLP) Neural Network. Inputs which are belong to NASA 93 dataset and Desharnais dataset are trained by using Levenberg-Marquardt backpropagation with 20 hidden neurons for 1000 epochs. Skeleton of the networks are shown in Fig. 2.

Secondly, inputs are carried on ANFIS. Subractive clustering method is used in order to generate Fuzzy Inference System with the features that range of influence is 0.5, squash factor 1.25, accept ratio 0.5 and reject ratio 0.15. System is trained by using backpropagation optimization method with no error tolerance for 1000 epochs. Structure of models for NASA 93 and Desharnais dataset are shown in Fig. 3 and Fig. 4. As it is seen in the figures, 41 rules for Desharnais dataset and 17 rules for NASA 93 datasets are created randomly by ANFIS.

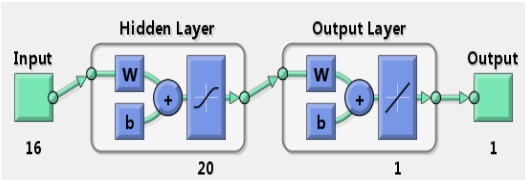

Fig. 2. Skeleton of the networks.

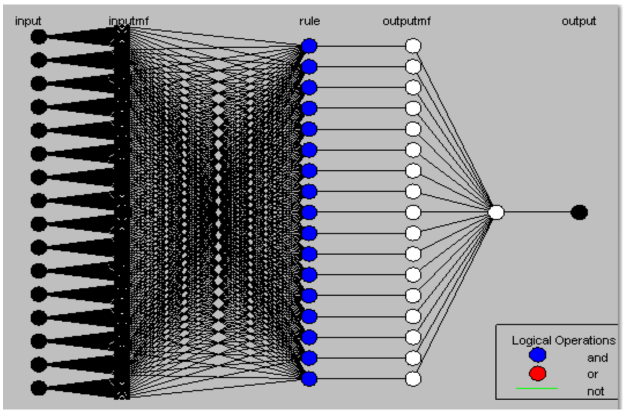

Fig. 3. Structure of ANFIS for NASA 93 dataset.

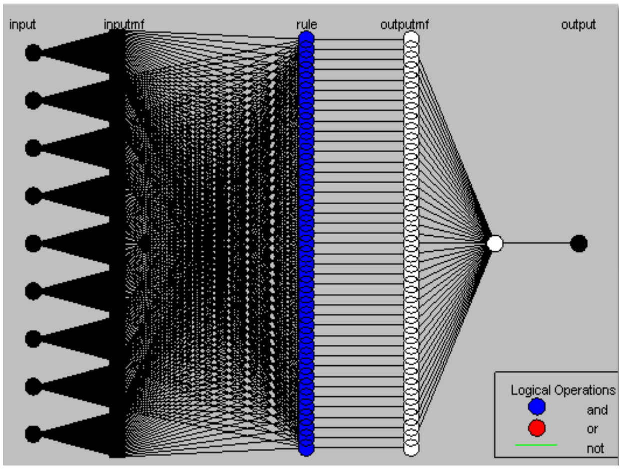

Fig. 4. Structure of ANFIS for Desharnais dataset.

IV. Results and Discussion

In this study, prediction of software estimation is done by using Multilayer Perceptron and Adaptive Neuro Fuzzy Inference System. Effort of 10 projects which are belong to NASA 93 dataset and effort of 11 projects which are belong to Desharnais dataset are predicted. Predictions and performance measurements for NASA 93 Projects are shown in Table I and Table II.

TABLE I: Predicted Efforts Calculated by MLP and Performance Measurements for NASA 93 Projects

Actual Effort	Predicted Effort Calculated by MLP	MRE
117.6	176.9062	0.5043
117.6	73.7686	0.3727
31.2	22.816	0.2687
36	24.2996	0.3250
25.2	28.7591	0.1412
10.8	10.4115	0.0359
352.8	207.483	0.4119
192	226.8248	0.1814
18	16.3055	0.0941
50	30.8447	0.3831

TABLE II: Predicted Efforts Calculated by ANFIS and Performance Measurements for NASA 93 Projects

Actual Effort	Predicted Effort Calculated by ANFIS	MRE
117.6	116.9909	0.0052
117.6	110.5151	0.0060
31.2	26.3274	0.1561
36	28.8181	0.1995
25.2	36.2904	0.4401
10.8	5.405	0.4995
352.8	319.7385	0.0937
192	177.1587	0.0773
18	15.3681	0.1462
50	39.7775	0.2045

Predictions and performance measurements for Desharnais dataset are shown in Table III and Table IV.

TABLE III: Predicted Efforts Calculated by MLP and Performance Measurements for Desharnais Projects

Actual Effort	Predicted Effort Calculated by MLP	MRE
1155	2332.6	1.0196
546	783.6765	0.4353
2275	3046	0.3389
9100	23467	1.5788
595	816.5221	0.3723
13860	13984	0.0089
1400	6988.2	3.9916
2800	8228.8	1.9386
9520	23082	1.4246
5880	17368	1.9537
23940	35512	0.4833

The best performance result in effort estimation is lower MMRE and higher PRED (0.25) . As it is seen in Table V, predictions of ANFIS for both NASA 93 and Desharnais Projects are more close to the actual outputs with lower

MMRE and higher Pred (0.25) than MLP. Thus, it can be said that prediction of Adaptive Neuro Fuzzy System is better than Multilayer Perceptron.

TABLE IV: PREDICTED EFFORTS CALCULATED BY ANFIS AND PERFORMANCE MEASUREMENTS FOR DESHARNAIS PROJECTS

Actual Effort	Predicted Effort Calculated by ANFIS	MRE
1155	2457.8	1.128
546	567.7534	0.0398
2275	3288.4	0.4455
9100	18054	0.984
595	1220.2	1.0508
13860	14373	0.0370
1400	3748.4	1.6774
2800	8512	2.04
9520	13449	0.4127
5880	15777	1.6832
23940	3343.3	0.8603

TABLE V: RESULTS SUMMARY FOR TWO TECHNIQUES

Datasets	MLP		ANFIS	
	MMRE	PRED(0.25)	MMRE	PRED(0.25)
NASA 93	0.27	0.4	0.18	0.8
Desharnais	1.23	0.09	0.94	0.18

Results summary for two datasets and two techniques is given in Table V.

In addition, it is observed that predictions for NASA 93 Projects are more accurate than Desharnais Projects. In this study, the best performance is belongs to ANFIS. However, using ANFIS, MMRE and Pred (0.25) values for NASA 93 Projects are 0.18 and 0.8 whereas these values for Desharnais Projects are 0.94 and 0.18. It means that some classifications must be done for Desharnais dataset in order to get better performance and better accuracy.

V. CONCLUSIONS

In this study, software effort estimation is predicted by using Multilayer Perceptron and Adaptive Neuro Fuzzy Inference System.

NASA 93 and Desharnais datasets are used for training and testing the system. MMRE and PRED (0.25) are performance evaluation criterions.

At the end of the experiment, we observed that performance of ANFIS is better with lower MMRE and higher PRED (0.25) for both NASA 93 and Desharnais Projects. For example; while MMRE and Pred (0.25) values of ANFIS for NASA 93 Projects are 0.18 and 0.8, MLP values are 0.27 and 0.4. On the other hand, we found that MMRE and Pred (0.25) values of ANFIS for Desharnais Projects are 0.94 and 0.18, these values are 1.23 and 0.09 when MLP is used. As a result, it can be said that performance of ANFIS is better than performace of MLP when NASA 93 and Desharnais datasets are used as a sample and MMRE and Pred (0.25) are used as a performance criterion. In addition, it is observed that predictions for NASA 93 Projectes are more close to actual efforts when compared predictions for Desharnais Projects.

As a future work, it is planned to make some classification on inputs in order to improve performance of effort estimation and use fuzzy logic to predict software effort.

REFERENCES

[1] G. Batra, "An intelligent algorithm for software effort estimation," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, issue 5, May 2012.

[2] R. Malhotra and A. Jain, "Software effort prediction using statistical and machine learning methods," *International Journal of Advanced Computer Science and Applications*, vol. 2, no.1, pp. 145-152, January 2011.

[3] N. Garcia-Diaz, C. Lopez-Martin, and A. Chavoya, "A comparative study of two fuzzy logic models for software development effort estimation," in *Proc. the 2013 Iberoamerican Conference on Electronics Engineering and Computer Science*, pp. 305-314, Procedia Technology 7, 2013.

[4] C. Lopez-Martin, "A fuzzy logic model for predicting the development effort of short scale programs based upon two independent variables," *Applied Soft Computing*, vol. 11, pp. 724-732, 2011.

[5] M. W. Gardner and S. R. Dorling, "Artificial neural networks (The multilayer perceptron) - A review of applications in the atmospheric sciences," *Atmospheric Environment*, vol. 32, no. 14/15, pp. 2627-2636, 1998.

[6] A. Khadema and G.-A. Hossein-Zadeh, "Estimation of direct nonlinear effective connectivity using information theory and multilayer perceptron," *Journal of Neuroscience Methods*, vol. 229, pp. 53-67, 2014.

[7] J.-S. R. Jang, "ANFIS: Adaptive-network-based fuzzy ınference system," *IEEE Transactıons on Systems, Man, and Cybernetıcs*, vol. 23, no. 3, May/June 1993.

[8] R. Singh, A. Kainthola, and T. N. Singh, "Estimation of elastic constant of rocks using an ANFIS approach," *Applied Soft Computing*, vol. 12, pp. 40-45, 2012.

[9] B. Khoshnevisan, S. Rafiee, M. Omid, H. Mousazadeh, and S. Clark, "Environmental impact assessment of tomato and cucumber cultivation in greenhouses using life cycle assessment and adaptive neuro-fuzzy inference system," *Journal of Cleaner Production*, vol. 73, pp. 183-192, 2014.

[10] B. B. Ekici and U. T. Aksoy, "Prediction of building energy needs in early stage of design by using ANFIS," *Expert Systems with Applications*, vol. 38, pp. 5352–5358, 2011.

[11] J. S. Shirabad and T. J. Menzies. (2005). The PROMISE Repository of Software Engineering Databases. School of Information Technology and Engineering, University of Ottawa, Canada. [Online]. Available: http://www.promise.site.uottawa.ca/SERepository

Berna Seref received her B.Sc. in computer engineering from Anadolu University, Eskisehir, Turkey in 2012. She is currently a M. Sc. computer engineering student at Kirikkale University, Kirikkale and she is working as a research assistant at Dumlupinar University, Kutahya, Turkey. Her interest areas are artificial intelligence, database management systems and computer programming.

Necaattin Barisci graduated from Gazi University in 1994. He took his M.Sc. and Ph.D degrees from Gazi University in 2000 and 2005, respectively, all in electronic and computer technology. He is an associate professor at the Department of Computer Engineering at Gazi University. His interest areas are biomedical computing, artificial intelligence and software. He has written numerous articles on related areas.