Using COCOMO Dataset Effort Estimation for Developing Software

Manohar K. Kodmelwar, S.D. Joshi, V. Khanna

Abstract--- The development of software according to the required demand of customer. So it is important to find the effort required to develop the software. The estimation is done using the particle swarm optimization technique by applying the weights in the neural network. The usage of NN helps in estimating the efforts of the software with less cost and failure rates. The NN is utilized together with optimization to attain an enhanced outcome. The dataset utilized as input to the proposed system is the COCOMO dataset. The actual effort in the COCOMO dataset is measured by person-month which represents the number of months that one person needs to develop a given project. The proposed method gives the accurate estimation compared to the existing models

Index Terms--- COCOMO, NN, Hybrid swarm particle optimization, credit assignment path.

I. INTRODUCTION

The creation of Software is a complex procedure. Proper requirement from customer, planning, executing the task. Due to the changing the requirement of society the adaptation of change and developing the changes are at most important. The number of developing methods are exists to for developing the software. The standardized methods are available in terms of regulation and policies to develop the software according to the requirement of customer. The software development methods like waterfall model, RAD, incremental model and agile models are available. According to the type of project scenario the methods are selected for development.

The project manager collects the information from the business analysts, prepare his team accruing to the project, gives the detail idea about project and develops the software.

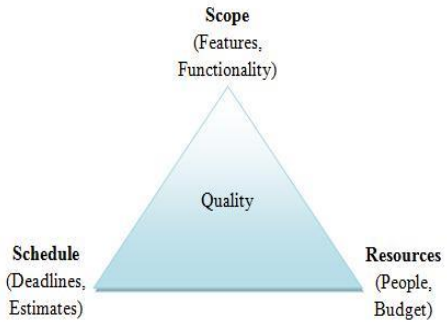

Fig. 1.1: Project Equilibrium Triangle

The software developed is testes at the developer side and customer side for to deploy properly the project. To develop the software the equilibrium triangle will help to check the progress of project shown in fig 1.1.

According to the proposed method of Magne Jørgensen to bring more practicality in the judgment-based estimates of work effort. He formed the three groups of experts, the one group use the proposed method second group use the traditional approach and third group use the probabilistic effort estimation method. By observation he found the proposed method is more realistic compared traditional methods.

The method of expert judgments proposed by the Tony Rosqvist et. al [6] to find the software quality. The individual opinion about the various factors of quality considered and finally the group decision was taken into consideration for the overall the number of groups. The method act as good supporting tool for quality of software.

The particle swarm optimization techniques proposed by the Mandeep Kaur and Sumeet Kaur Sehra [7] using the function point analysis. By using value adjustment factor to the function point analysis optimized estimation can be done.

The proposed technique by Valentina Lenarduzzi et.al [9] to use the functional size metric to the scrum. By this he put the idea of estimate in agile methodology scrum.

The general frame work for effort estimation by giving the scope and priorities to the estimation model to check with historical data and estimating the cost, schedule and features. Shown in fig 1.2

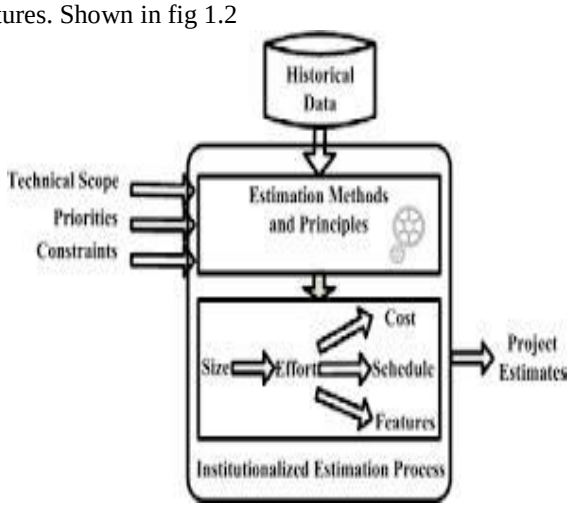

Fig. 1.2: General framework for effort estimation

Revised Version Manuscript Received on 22 February, 2019

Manohar K. Kodmelwar, Research Scholar, Bharath University, Chennai, India.

S.D. Joshi, Faculty of Engineering and Technology, Bhartividyapth, Pune, India.

V. Khanna, Department of Information Technology, Bharath University, Chennai, India.

II. PROPOSED METHODOLOGY

In proposed methodology the COCOMO dataset is used as input he deep neural network by adjusting the weights. The output is fed to the optimization block for hybrid swarm particle optimization.

The algorithm for particle swarm optimization is as follows Begin

for each particle

Initialize the position and velocity

end for.

do

for each particle

Calculate fitness value of the particle *F*

if *F* > *pbest*

Set *pbest* = current fitness value end if

if *pbest* > *gbest*

Set *gbest* = Best fitness value of all particles end if

for each particle

Calculate particle velocity

Update particle position

for each particle

Apply cross over

and

Mutation

end for

end for

end for

end

The deep learning consist of number of layers , each layer passes the data to next layer. The credit assignment path gives the connected chain of number of layers. CAP gives the idea of how the number of layers are connected. The intermediate layers are also called as a the hidden layers in the NN. The fig 2.1 shows the deep learning using number of hidden layer

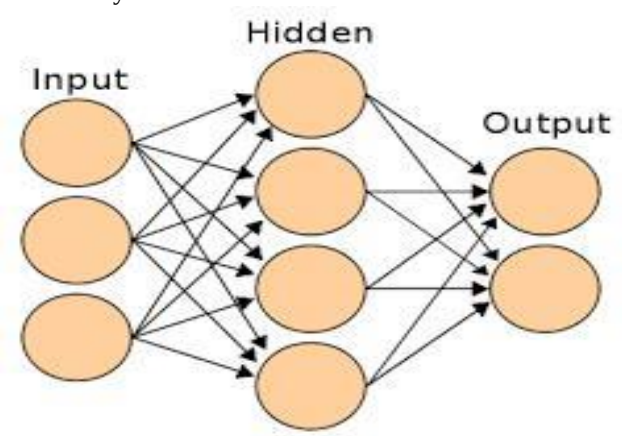

Fig. 2.1: Frame work for Neural Network

The proposed algorithm uses the formula for finding the mean magnitude of the relative error by the following equation for neural network

$$MAPE = \frac{1}{N}\sum_{\alpha=1}^{N}\left|\frac{Εστιματεδ - Αχτυαλ}{Αχτυαλ}\right|$$

Της Μεαν Αβσολυτε Ερρορ (MAE) φουνδ βψ

$$MAE = \frac{1}{\nu}\sum_{\iota=1}^{\nu}\left|\psi_\iota - \psi_\iota\right|$$

averaging over the test sample between prediction and actual observation.

III. CONCLUSION

This study proposed a NN based approach to deal with accuracy in the effort estimation technique. For improving the performance of NN based approach the hybrid optimization In order to evaluate the performance of the NN based approach, we used a hybrid optimization technique. The COCOMO dataset used as input to the NN. The proposed approach gives the best result compared to traditional effort estimation methods.

IV. FUTURE SCOPE

In the future work analogy based estimation used with the combination of neural network to get the best results for software estimation. The analogy based estimation will add the additional input for improving the effort estimation.

REFERENCES

1. Karnavel, K., and R. Dillibabu, "Development and application of new quality model for software projects", The Scientific World Journal, Vol. 2014, 2014.
2. Saleem Basha, and Dhavachelvan Ponnurangam, "Analysis of empirical software effort estimation models", 2010.
3. Ashish Sharma, and Dharmender Singh Kushwaha, "Estimation of software development effort from requirements based complexity", Procedia Technology, 2012, Vol. 4, PP. 716-722.
4. Magne Jørgensen, "Contrasting ideal and realistic conditions as a means to improve judgment-based software development effort estimation", Information and Software Technology, 2011, Vol. 53, No. 12, PP. 1382-1390.

5. Philip Morrow, F. George Wilkie, and I. R. McChesney, "Function point analysis using NESMA: simplifying the sizing without simplifying the size", Software Quality Journal, 2014, Vol. 22, No. 4, PP. 611-660.
6. Tony Rosqvist, Mika Koskela, and Hannu Harju, "Software quality evaluation based on expert judgement", Software Quality Journal, 2003, Vol. 11, No. 1, PP. 39-55.
7. Mandeep Kaur, and Sumeet Kaur Sehra, "Particle swarm optimization based effort estimation using Function Point analysis", In Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014 International Conference on, IEEE, 2014, PP. 140-145.
8. Frank Vijay, J., "Enrichment of accurate software effort estimation using fuzzy-based function point analysis in business data analytics", Neural Computing and Applications, 2018, PP. 1-7.
9. Valentina Lenarduzzi, Ilaria Lunesu, Martina Matta, and Davide Taibi, "Functional size measures and effort estimation in agile development: a replicated study", In International Conference on Agile Software Development. Springer, Cham, 2015, PP. 105-116
10. Dengsheng Wu, Jianping Li, and Chunbing Bao, "Case-based reasoning with optimized weight derived by particle swarm optimization for software effort estimation", Soft Computing, 2018, Vol. 22, No. 16, PP. 5299-5310.

ACKNOWLEDGEMENT

I am thankful to the Dr. S. D. Joshi for encouraging & helping in doing Ph.D. I thankful to Dr. V. Khanna for guiding me during th software development. I am also thankful to my family for supporting. I am thankful to all friends .

AUTHOR DETAILS

Manoahr K. Kodmelwar is a research scholar at Bharat university Chennai, Tamilnadu. He completed the Engineering Degree & Master of Computer Engineering. He is interested in software engineering and operating system domain.

Dr.S.D. Joshi completed the PhD. From bharati vidyapeth. His area of intrest is software engineering. He the Head of Dept. Of Information Technology in bharati vidyapeth Pune.

Dr.V. Khanna is the dean if information technology at Bharath University Chennai, Tamilnadu. His area of interest is software engineering.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication