Software Cost Estimation using Particle Swarm Optimization and Neural Network

Sheena Goyal[#1], Sheilly Padda[*2]
[#]Student (Pursuing M.Tech in CSE), [*]Assistant Professor, Department of computer science and engineering, Chandigarh Group of colleges, Landran, Mohali, Punjab, India

Abstract – *In today's time, cost estimation of the project is the major part. Whenever a project is required by any company or organization, then the second major thing along with project requirements comes the cost they can spent on the project. In project management, planning and resources are allocated according to the cost it takes. Only then resources are described and assigned which includes the effort in allocating the resources and time in allocation of resources. In our research we are going to minimize the effort estimation non-linearity using clustering technique called K-means clustering. Then parameters of each cluster are tuned using PSO. These clusters and tuned parameters are further trained using back propagation.*

Keywords – *PSO, Software cost estimation, K-means clustering.*

I.INTRODUCTION

For the software industry, software cost estimation has been a challenge from a long time. Accurate cost estimation helps software companies making correct cost estimation and avoiding loss of money due to the wrong cost estimation. Software cost estimation can be defined as the estimated judgment of the expenses on the resources and the effort required allocating the resources for an undertaking. Cost estimation is typically measured regarding effort. The most widely recognized metric used is individual months or years (or man months or years). The effort is the measure of time for one individual to work for a certain time on the project. It is essential to record the attributes of a particular in the advancement environment into light when contrasting the effort of two or more undertakings in light of the fact that no two situations that are improved are same.

II.LITERATURE REVIEW

G. Sivanageswara Rao, Ch.V. Pani Krishna and K. RajasekharaRao et al (2013) [1] worked on COCOMO and MARE and used Multi Objective PSO to tune the parameters. The parameters of model tuned by using MOPSO considered two objectives, Mean Absolute Relative Error and Prediction. They observed that the model they have proposed had given better results when compared with the standard COCOMO model.

Ali Idri et al [10] stated that web Effort Estimation is a procedure of anticipating the endeavors and cost regarding cash, timetable and staff for any product venture framework. Numerous estimation models have been proposed in the course of the most recent three decades and it is accepted that it is an absolute necessity with the end goal of: Budgeting, danger examination, venture arranging and control, and task change speculation investigation.

Haigang Li et al (2015) [12] demonstrated that PSO calculation is a smart streamlining calculation in view of swarm knowledge. Molecule swarm enhancement calculation is straightforward, simple to execute, and it has a wide application prospect in investigative research and building applications. In actuality, most of the enhancement issue is the advancement issue of some nonlinear discrete with the presence of neighborhood. PSO calculation additionally has a few deformities in treating advancement issue. The ideal execution of the PSO calculation is proficiency; the quality weights are streamlined, which is the same as to enhance the precision of case recovery. The utilization of case is based thinking in the streamlining of weight vessel model outline. Through the examination comes about, the improvement of the execution of PSO calculation is better; the consequence of forecast is more rough to the real esteem, which can address the issues of reasonable applications in designing.

III.SOFTWARE COST ESTIMATION

All people in software industry know that there are lots of uncertainties involved in software cost estimation. So, it is difficult to expect perfect effort estimates even in perfect estimation process. The software development effort of a software project is more frequently estimated by project managers using the estimated effort they calculate the cost and duration associated with the project. Accurate development effort estimation at the earlier stage of a software development cycle is the most important to plan, monitor and control the allocated resources appropriately.

The uncertainties and factors on which software development effort depends on the following factors:

• The amount of Implemented functionality [11]

• Number of Errors made by programmers [11]
• Quality of code produced [11]
• Availability of development tools [11]
• Probabilistic factors (absence of staff members due to sickness) [11]
• Availability of skilled persons [11].

IV.COCOMO MODEL

COCOMO stands for Constructive Cost Model; it is a software cost estimation model that was first published in 1981 by Barry Bohem. It is an algorithmic approach to estimating the cost of a software project. By using COCOMO you can calculate the amount of effort and the time schedule for projects. From these calculations you can then find out how much staffing is required to complete a project on time. COCOMO's main metric used for calculating these values is lines of code (denoted KLOC for COCOMO II, or KDSI for COCOMO 81 and measured in thousands), function points (FP), or object points (OP). COCOMO 81 was the first form of COCOMO. It has been discovered that generally speaking it has the capacity produce gauges that are inside 20% of the real values 68% of the time.

COCOMO model has three different modes

1. Organic
2. Embedded
3. Semi detached

V. PSO (PARTICLE SWARM OPTIMIZATION)

PSO is a stochastic global optimization method which is based on simulation of social behavior. As in GA and ES, PSO exploits a population of potential solutions to probe the search space. In contrast to the aforementioned methods in PSO no operators inspired by natural evolution are applied to extract a new generation of candidate solutions. Instead of mutation PSO relies on the exchange of information between individuals, called particles, of the population, called swarm. In effect, each particle adjusts its trajectory towards its own previous best position, and towards the best previous position attained by any member of its neighborhood. In the global variant of PSO, the whole swarm is considered as the neighborhood. Thus, global sharing of information takes place and particles profit from the discoveries and previous experience of all other companions during the search for promising regions of the landscape. To visualize the operation of the method consider the case of the single objective minimization case; promising regions in this case possess lower function values compared to others, visited previously [3].

1. ALGORITHM OF PSO

Numbers of birds are searching for food randomly in an area. There is only one piece of food is available. All birds do not know where the food is, but they know how far it is in iteration. So the best solution to find out the food is to follow the bird nearest to the food.

PSO is used for optimization problem. Each single solution in the search space is a bird. We call it particle.

All the particles have

1. Fitness values, evaluated by fitness function
2. Velocities which direct the flying of the object

Particles fly through the problem space by following current optimization particles.

PSO is initialized with a group of random particles (solutions) and searches for optima by updating generations.

In each iteration, each particle is updated by following two “best” values. The first solution is the best solution it has achieved so far. The value is called “pbest”. Another best value that is tracked by PSO is the best value obtained by any particle in the population. This best value is a global best and called “gbest”. When a particle takes part of the population as its topological neighbors, the best value is a local best and is called “lbest”.

After finding the two best values, the particle updates its velocity and positions with following equation (a) and (b).

v[ ]=v[ ] + c1 * rand( ) * (pbest[ ] - present[ ]) + c2 * rand( ) * (gbest[ ] - present[ ])(a)

present[ ] = present[ ] + v[ ] (b)

v[ ] is the particle velocity, present[ ] is the current particle (solution). pbest[ ] and gbest[ ] are defined as stated before. rand( ) is a random number between (0, 1). c1, c2 are learning factors. Usually c1 = c2 = 2.

VI. BACK PROPAGATION ALGORITHM

The back propagation learning algorithm is one of the most widely used methods in neural network. The network associated with back-propagation learning algorithm is called as back propagation network. While training a network a set of input-output pair is provided the algorithm provides a procedure for changing the weight in BPN that helps to classify the input output pair correctly. Gradient descent method of weight updating is used.

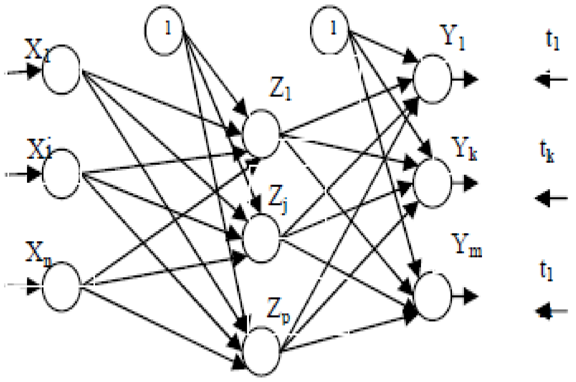

Figure: Architecture of a Back propagation Network

The aim of the neural network is to train the network to achieve a balance between the ability of net to respond and its ability to give reasonable responses to the input that is similar but not identical to the one that is used in training. Back propagation algorithm differs from the other algorithm by the method of weight calculation during learning. The drawback of Back propagation algorithm is that if the hidden layer increases the network become too complex [11].

VII.CONCLUSION AND FUTURE SCOPE

In our research we are going to minimize the effort estimation non-linearity using clustering technique called K-means clustering. Then parameters of each cluster are tuned using PSO. These clusters and tuned parameters are further trained using back propagation. We are using clustering techniques as well as back propagation neural network algorithm as drivers for the improvement. For further work, any genetic algorithm can be used for improving the linearity of the results.

VIII.REFRENCES

[1] G. Sivanageswara Rao, Ch.V. Pani Krishna and K. RajasekharaRao. Multi objective Swarm Optimization for cost estimation. Advances in intelligent systems and computing 248, convention of CSI, volume I

[2] Jun-qing Li, Quan-ke Pan, Sheng-xianXie, Bao-xianJia and Yu-ting Wang (2010). A hybrid particle swarm optimization and tabu search algorithm for flexible job-shop scheduling problem. International Journal of Computer Theory and Engineering, Vol. 2, No. 2, 1793-8201

[3] Konstantinos E. Parsopoulos and Michael N. Vrahatis. Particle Swarm Optimization Method for Constrained Optimization Problems

[4] DiptamDutta, Argha Roy, KaustavChaudhury (2014). Training Artificial Neural Network using Particle Swarm Optimization Algorithm. International Journal of Advanced Research in Computer Science and Software Engineering Volume 4, Issue 5, May 2014 ISSN: 2277 128X

[5] TaghiJavdani Gandomani, KohTieng Wei and Abdulelah Khaled Binhamid (2014). A Case Study Research on Software Cost Estimation Using Experts' Estimates, Wideband Delphi, and Planning Poker Technique. International Journal of Software Engineering and Its Applications Vol. 8, No. 11, pp. 173-182

[6] Poonam Rijwani, Sonal Jain and Dharmesh Santani (2014).Software Effort Estimation: A Comparison Based Perspective. International Journal of Application or Innovation in Engineering & Management (IJAIEM), Volume 3, Issue 12, ISSN 2319 - 4847

[7] Ali Arshad (2014). A Survey on Software Cost Estimation in the Pakistani Software Industry by International Journal of Computer and Electronics Research [Volume 3, Issue 1, February 2014]

[8] Mudasir Manzoor Kirmani and Abdul Wahid (2015). Use Case Point Method of Software Effort Estimation: A Review. International Journal of Computer Applications (0975 – 8887) Volume 116 – No. 15

[9] Mohammad Islam1, Dr. Vinodani Katiyar (2014). Development of a software maintenance cost estimation model: 4TH GL PERSPECTIVE. International Journal of Technical Research and Applications e-ISSN: 2320-8163, Volume 2, Issue 6, PP. 65-68

[10] Ali Idri. A Fuzzy Decision Tree to Estimate Development Effort for Web Applications. International Journal of Advanced Computer Science and Applications, Special Issue on Artificial Intelligence

[11] E. Praynlin, P. Latha (2014). Software Effort Estimation Models Using Radial Basis Function Network. World Academy of Science, Engineering and Technology, International Journal of Computer, Control, Quantum and Information Engineering Vol:8, No:1

[12] Haigang Li, Qian Zhang and Yong Zhang (2015). Improvement and Application of Particle Swarm Optimization Algorithm based on the Parameters and the Strategy of Co-Evolution. Applied Mathematics & Information Sciences An International Journal, Appl. Math. Inf. Sci. **9**, No. 3, 1355-1364

[13] Farooq Azam, Salman Qadri, Shabir Ahmad, Kamran Khan, Abu Buker Siddique, Bilal Ehsan (2014). Framework Of Software Cost Estimation By Using Object Orientated Design Approach. International Journal of Scientific & Technology Research Volume 3, Issue 8, ISSN 2277-8616

[14] Swati Waghmode, Dr.Kishor Kolhe (2014). A Novel Way of Cost Estimation in Software Project Development Based on Clustering Techniques. International Journal of Innovative Research in Computer and Communication Engineering Vol. 2, Issue 4

[15] Isa Maleki, Laya Ebrahimi, Saman Jodati, Iraj Ramesh (2014). Analysis of software cost estimation using fuzzy logic. International Journal in Foundations of Computer Science & Technology (IJFCST), Vol.4, No.3, May 2014

[16] Barry Boehma, Chris Abts and Sunita Chulani (2000). Software development cost estimation approaches –A survey

[17] Adanma C. Eberendu (2014).Software Project Cost Estimation: Issues, Problems and Possible Solutions.

International Journal of Engineering Science Invention ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726, Volume 3 Issue 6ǁ June 2014 ǁ PP.38-43

[18] Omprakash Tailor, Jyoti Saini, Mrs. Poonam Rijwani (2014). Comparative analysis of software cost and effort estimation methods: A Review. International Journal of Computer Science and Mobile Computing, Vol.3 Issue.4, April- 2014, pg. 1364-1374, ISSN 2320–088X

[19] Dr. S. Malathi, B. S. Lijin (2014), an Efficient Method for the Estimation of Effort in Software Cost. International Journal of Advance Research in Computer Science and Management Studies, Volume 2, Issue 2, February 2014, ISSN: 2321-7782 (Online)

[20] Farhad Soleimanian Gharehchopogh, Awat Maroufi (2014). Approach of software cost estimation with hybrid of imperialist competitive and artificial neural network algorithms. Journal of Scientific Research and Development 1 (1): 50-57, 2014, ISSN 1115-7569

[21] Farhad Soleimanian Gharehchopogh, Isa Maleki, Amin Kamalinia, Habibeh Mohammad Zadeh (2014). Artificial bee colony based constructive cost model for software cost estimation by Journal of Scientific Research and Development 1 (2): 44-51, 2014, ISSN 1115-7569

[22] Peram Subba Rao, K. Venkata Rao, P. Suresh Varma and Tai-hoon Kim (2014). Programming Evaluation Process Using Hybrid Cost Estimation Model by International Journal of Software Engineering and Its Applications Vol. 8, No. 10, pp. 55-64

[23] Abdu Gumaei, Bandar Almaslukh, Nejmeddine Tagoug (2015). An Empirical Study of Software Cost Estimation in Saudi Arabia Software Industry. International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307, Volume-4 Issue-6, January 2015

[24] Jairus Hihn, Tim Menzies, Naveen Lekkalapudi, James Johnson (2014) The NASA Software Cost Model: Formalizing analogy based cost estimation, October 21-23, 29th International Forum on COCOMO and Systems Software Cost Modeling