Software Cost Estimation using Fuzzy Logic Technique

Ravneet Preet Singh Bedi and Amardeep Singh
Department of Computer Engineering, Punjabi University, Patiala – 147002, Punjab, India; bedirps2000@yahoo.com, amardeep_dhiman@yahoo.com

Abstract

Estimation of software development cost has been a challenging research area. Soft computing based techniques such as fuzzy logic outperform traditionally used methods in terms of accuracy of estimation. The current research presents a novel method that shows promising results. The results are compared with COCOMO technique and the accuracy level is improved considerably. The proposed method is simple yet effective as it implements the technique using MATLAB’s fuzzy logic toolbox.

Keywords: COCOMO, Efforts, Fuzzy Logic, Software

1. Introduction

Software engineering research deals with various aspects of software development. Cost estimation remains one of the most critical areas of research due to the financial aspects involved. Accuracy in terms of cost estimation is sought primarily because over pricing or under-pricing, both can hurt the enterprise that is developing the software. The special requirement of high accuracy leads the research from orthodox techniques like CoCoMo I and II etc. to more contemporary techniques such as neural network, data mining, genetic algorithms and neuro-fuzzy techniques.

One of the soft computing technique fuzzy logic is very popular in terms of prediction or estimation technique. There are a number of decision making applications available which employ fuzzy logic as a main tool. Fuzzy logic has been used for prediction in medical diagnosis, political predictions, sports and finance etc.

This research paper highlights the use of fuzzy logic to predict or estimate the cost of software development to a great accuracy. The dataset used for this research can be found at the Promise repository. It is a well-established set of records pertaining to the factors that affect the cost of software and the incurred cost, thereof.

The current section of this paper introduces the problem and solution technique. The remainder of this paper is organised as follows: The second section details the review of related literature, the third section provides an insight to the proposed technique, the fourth section deals with the results of extensive experiments carried out and the fifth section concludes this discourse.

2. Literature Review

Du et al. combine neuro-fuzzy model with SEER-SEM technique to achieve an accuracy that is claimed to be 18% higher than its counterparts. Dizaji and Gharehchopogh propose a technique which a unique blend of ant colony optimization and chaos optimization algorithms to control the mae to 0.078 as compared to 0.29 for COCOMO. Shivakumar device a nuero-fuzzy technique to estimates software development efforts and are able to show promising results. But, the proposed technique is not very suitable to large datasets. Literature shows the use of augmented fuzzy logic for the purpose of estimation of software development efforts. The work by Sharma and Verma uses the Gaussian MFs and produce excellent results by managing the inaccuracy in inputs very well. Also, their quality to adopt further helps them a right
Software Cost Estimation using Fuzzy Logic Technique

3. Proposed Technique

Soft computing is an area of research that deals with real life problems in a more effective way, thus providing more accurate results. This proposed work is based on using Fuzzy Logic (FL) based technique to predict efforts to be spent on a given software development project. Figure 1 shows the proposed model used for estimation based on FL. The fuzzy inference system that is proposed in this research work is based on Mamdani system. The model requires five input parameters viz. Complexity, Data, Tool, loc (lines of code) and skills. There is one output parameters named Estimate is used for the prediction of software efforts.

Table 1. Linguistic variables for input/output parameters.

| Input/output parameter | Linguistic variables used |
|------------------------|---------------------------|
| Complexity             | Simple, Less, Medium, High, Very High |
| Data                   | Free, Low, Average, High |
| Tool                   | Very Low, Low, Medium, High, Very High |
| Loc                    | Bare, Average, Very High |
| Skills                 | Novice, Average, Good, Expert |
| Estimate               | Low, Medium, High |

Based on the above linguistic variables, these input parameters are applied with four fuzzy rules. The fuzzy rules are defined as follows:
If (Complexity is Simple) and (Data is Free) and (Tool is low) and (loc is Bare) and (Skills is Avg) then (Estimated is low) (1)

If (Complexity is Less) and (Data is Low) and (Tool is Medium) and (loc is Average) and (Skills is Good) then (Estimated is High) (1)

If (Complexity is Medium) and (Data is Average) and (Tool is High) and (loc is VeryHigh) and (Skills is Expert) then (Estimated is High) (1)

If (Complexity is High) and (Data is High) and (Tool is VeryHigh) and (loc is Average) and (Skills is Good) then (Estimated is High) (1)

The above model is capable of utilising all three input factors and apply predefined fuzzy rule base to get an accurate prediction of software efforts. The results thus produced are compared with COCOMO II. COCOMO I & II lack the precision due to the reason that these models do not consider all input parameters especially the COCOMO I. The trouble with COCOMO II is that when applied to the records from Promise dataset, it tends to misinterpret, both over as well as under. Whereas, the proposed model when applied to the same dataset produce results that are very much aligned with the actual results given with the records.

Figure 2 shows a screenshot of proposed FIS developed using Matlab 2015a.

### 4. Experimental Results

Extensive experimentation has been done to assert the suitability of the proposed method as compared to existing methods found in the literature. The vast amount of data available through the dataset makes it relatively straightforward to experimentally analyse various available techniques. Table 2 shows the comparison of proposed technique using fuzzy logic and existing COCOMO technique for various software projects randomly chosen from the selected data set. The results are also depicted pictorially using graph. The proposed technique succeeds at giving accurate results (Figure 3).

**Table 2. Comparison of Effort Estimation Results in MRE.**

| Sl. No. | MRE (%) using COCOMO | MRE (%) using Proposed Technique |
|---------|-----------------------|----------------------------------|
| 1.      | 18.6170               | 15.2470                          |
| 2.      | 33.2460               | 25.6070                          |
| 3.      | 38.8352               | 11.1520                          |
| 4.      | 2.7500                | 0.4110                           |
| 5.      | 27.3928               | 15.9128                          |
| 6.      | 25.9296               | 19.5800                          |
| 7.      | 31.8756               | 1.0950                           |
| 8.      | 26.1911               | 21.9050                          |
| MMRE(%) | 25.6040               | 17.6130                          |

The percent MMRE (Mean Magnitude of Relative Error) shown as the last row of Table 2 reiterates the point made earlier. It shows that MMRE% of 17.613 as given by proposed technique is far superior to the MMRE% of 25.604 as given by well-established COCOMO technique.
5. Conclusion

Software development effort estimation remains an area of research since long. Even the well-established and widely used technique named COCOMO fails to show acceptable accuracy. Results show that there is a need to enhance the techniques to achieve acceptable accuracy. Soft computing based techniques show a great promise while calculating the estimates for efforts on software development. The result graphs show that the proposed technique result curve runs relatively very much closer to the actual curve as compared to the curve drawn for the existing technique. The results achieved using proposed technique encourage us to announce the suitability of the proposed technique for estimation of software development efforts in software companies.

6. References

1. Boehm BW. Software Engineering Economics, Prentice-Hall, Englewood Cliffs, NJ, 1994.
2. Zadeh LA. Fuzzy Sets, Information and Control. 1965; 8(3):338−53.
3. Mago, VK, Bhatia N, Bhatia A, Mago A. Clinical Decision Support System for Dental Treatment, Journal of Computational Science. 2012; 3(5):254−61.
4. Singh H, Singh G, Bhatia N. Election Results Prediction System based on Fuzzy Logic, International Journal of Computer Applications. 2012; 53(9).
5. Singh G, Bhatia N, Singh S. Fuzzy Logic Based Cricket Player Performance Evaluator, IJCA Special Issue on Artificial Intelligence Techniques-Novel Approaches and Practical Applications. 2011; 1(3):11−16.
6. Kumar S, Bhatia N, Kapoor N. Fuzzy logic based decision support system for loan risk assessment. In Proceedings of the International Conference on Advances in Computing and Artificial Intelligence, ACM. 2011, July; 179−82.
7. Du WL, Capretz LF, Nassif AB, Ho D. A Hybrid Intelligent Model for Software Cost Estimation, arXiv preprint arXiv:1512.00306.
8. Dizaji ZA, Gharehchopogh FS. A Hybrid of Ant Colony Optimization and Chaos Optimization Algorithms Approach for Software Cost Estimation, Indian Journal of Science and Technology. 2015; 8(2):128.
9. Shivakumar N, Balaji N, Ananthakumar K. A Neuro Fuzzy Algorithm to Compute Software Effort Estimation, Global Journal of Computer Science and Technology. 2016; 16(1).
10. Sharma V, Verma HK. Optimized Fuzzy Logic Based Framework for Effort Estimation in Software Development, International Journal of Computer Science Issues. 2010; 7(2):30−38.
11. Bhatnagar R, Ghose MK. Comparing Soft Computing Techniques for Early Stage Software Development Effort Estimations, International Journal of Software Engineering and Application. 2012; 3(2):119−27.
12. Sehra SK, Brar YS, Kaur N. Soft Computing Techniques for Software Project Effort Estimation, International Journal of Advanced Computer and Mathematical Science. 2011; 2(3):160−67.
13. Kad S, Vinay C. Fuzzy Logic Based Framework for Software Development Effort Estimation, International Journal of Engineering Science. 2011; 1:3330−342.
14. Potdar SM, Puri M, Potdar MP. Literature Survey on Algorithmic Methods for Software Development Cost Estimation, International Journal of Computer Technology and Application. 2014; 5(2):183−88.
15. Reddy CS, Raju K. A Concise Neural Network Model for Estimating Software Effort, International Journal of Recent Trends in Engineering. 2009; 1(1):183−93.
16. Shradhanand AK, Satbir J. Use of Fuzzy Logic in Software Development, Issues in Information Systems. 2007; 8(2).
17. Kumar S, Chopra V. To Design and Implement Neural Network and Fuzzy Logic for Software Development Effort Prediction, International Journal of Computer Applications. 2013; 84(11).
18. Boetticher G, Menzies T, Ostrand T. PROMISE Repository of Empirical Software Engineering Data, West Virginia University, Department of Computer Science. 2007. http://promisedata.org/ repository.
19. Dejaeger K, Verbeke W, Martens D, Baesens B. Data Mining Techniques for Software Effort Estimation: A Comparative Study, IEEE Transactions on Software Engineering. 2012; 38(2):375−97.
20. Oliveira AL, Braga PL, Lima RM, Côrnelio ML. GA-Based Method for Feature Selection and Parameters Optimization for Machine Learning Regression Applied to Software Effort Estimation, Information and Software Technology. 2010; 52(11):1155−66.
21. Azzeh M, Neagu D, Cowling PI. Fuzzy Grey Relational Analysis for Software Effort Estimation, Empirical Software Engineering. 2010; 15(1):60−90.
22. Bardsiri VK, Jawawi DNA, Hashim SZM, Khatibi E. A PSO-Based Model to Increase the Accuracy of Software Development Effort Estimation, Software Quality Journal. 2013; 21(3):501−26.
23. Ochodek M, Nawrocki J, Kwarciak K. Simplifying Effort Estimation Based on Use Case Points, Information and Software Technology. 2011; 53(3):200−13.
24. Azzeh M. A Replicated Assessment and Comparison of Adaptation Techniques for Analogy-based Effort Estimation, Empirical Software Engineering. 2012; 17(1-2):90–127.

25. Corazza A, Di Martino S, Ferrucci F, Gravino C, Sarro F, and Mendes E. Using Tabu Search to Configure Support Vector Regression for Effort Estimation, Empirical Software Engineering. 2013; 18(3):506–46.

26. Song Q, Shepperd M. Predicting Software Project Effort: A Grey Relational Analysis Based Method, Expert Systems with Applications, 2011; 38(6):7302–16.

27. Idri A, Azzahra Amazal F, Abran A. Analogy-Based Software Development Effort Estimation: A Systematic Mapping and Review, Information and Software Technology. 2015; 58:206–23.