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 Gharachopogh 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 neuro-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
selection for representing fuzzy sets. The framework is optimized in terms of flexibility in modifiable fuzzy sets as per the environment. Bhatnagar and Ghose\(^\text{11}\) developed a new method that present alternatives for approximation of the software development attempts, and in particular, the Computational Intelligence (CI) that moots the mechanisms of communication between humans and processes related information with the set aim of creating the Intelligent Systems (IS). Linear Regression Neural network has the smallest value for MMRE out of other existing models. When compared with the fuzzy logic, it has been established that the fuzzy logic performs better than the neural network models because of its lowest MMRE value. Sehra et al.\(^\text{12}\) have developed numerous models for the approximation. But there is none approximation method that can give the best estimates in different situations and every may be right choice in a special project judgment but there is no evaluation method which can current the best estimates in a variety of situations and each method can be appropriate in the individual scheme.

The plan of their work is to examine soft computing methods in the accessible models and to supply in depth evaluation of software and project judgment techniques accessible in industry and writing based on the unusual test datasets along with their power and weaknesses. Kad and Chopra\(^\text{13}\) research; developed the soft computing techniques to overcome the issues of uncertainty, partial availability. When the developer understand the problem of uncertainty and partial availability etc. the by using the technology of Fuzzy set like type-2, the researcher easily develop the system. Potdar et. al.\(^\text{14}\) research gives provided the brief introduction about the Algorithmic cost models and the various advantage and disadvantage related to this. It is not possible to say that which method is best or which is worst, because each method has its particular field where the developers use these methods. The choice of the method is not depend upon particular thing it depend upon the number of condition, it is better to select the combination of methods for the particular application. Reddy and Raju\(^\text{15}\) research is mainly developed the cost estimation that is based on the neural network. In their work they use the multilayer structure of the neural network. The whole structure is work on trained learning propagation method; the values that are produce from the training set are compares with the actual method to produce the correct results. Sharadan and et.al.\(^\text{16}\) research problems concern with the use of the fuzzy logic system, because fuzzy logic system are able to deal with the multi valued logic, uncertainty etc. but the simple system only concern with only two values 0 and 1. By using this type of approach the development process work like the human based system and easily deal with the multilayer architecture throughout the development of the software product. Kumar and Chopra\(^\text{17}\) present a comparative analysis between the fuzzy logic system and the neural network. A model which results lesser MMRE error is improved than that which gives higher MMRE error. Literature also suggests various other techniques such as based on data mining\(^\text{18}\), genetic algorithms\(^\text{19}\), fuzzy grey relational analysis\(^\text{20,21}\), PSO\(^\text{22}\), use case points\(^\text{23}\), analogy based\(^\text{24,25}\) and support vector regression\(^\text{26}\) to improve the accuracy of prediction of software efforts.

### 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. Based on the above linguistic variables, these input parameters are applied with four fuzzy rules. The fuzzy rules are defined as follows:

<table>
<thead>
<tr>
<th>Input/output parameter</th>
<th>Linguistic variables used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complexity</td>
<td>Simple, Less, Medium, High, Very High</td>
</tr>
<tr>
<td>Data</td>
<td>Free, Low, Average, High</td>
</tr>
<tr>
<td>Tool</td>
<td>Very Low, Low, Medium, High, Very High</td>
</tr>
<tr>
<td>Loc</td>
<td>Bare, Average, Very High</td>
</tr>
<tr>
<td>Skills</td>
<td>Novice, Average, Good, Expert</td>
</tr>
<tr>
<td>Estimate</td>
<td>Low, Medium, High</td>
</tr>
</tbody>
</table>

Table 1. Linguistic variables for input/output parameters.
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)

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\(^a\). 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.

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>MRE (%) using COCOMO</th>
<th>MRE (%) using Proposed Technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>18.6170</td>
<td>15.2470</td>
</tr>
<tr>
<td>2.</td>
<td>33.2460</td>
<td>25.6070</td>
</tr>
<tr>
<td>3.</td>
<td>38.8352</td>
<td>11.1520</td>
</tr>
<tr>
<td>4.</td>
<td>2.7500</td>
<td>0.4110</td>
</tr>
<tr>
<td>5.</td>
<td>27.3928</td>
<td>15.9128</td>
</tr>
<tr>
<td>6.</td>
<td>25.9296</td>
<td>19.5800</td>
</tr>
<tr>
<td>7.</td>
<td>31.8756</td>
<td>1.0950</td>
</tr>
<tr>
<td>8.</td>
<td>26.1911</td>
<td>21.9050</td>
</tr>
<tr>
<td>MMRE(%)</td>
<td>25.6040</td>
<td>17.6130</td>
</tr>
</tbody>
</table>

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.
The experiments show that the enhanced technique clearly outperforms the existing one.

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


