Abstract - Accuracy in the estimation of software Effort/Cost is one of the desirable criteria for any software cost estimation model. The estimation of effort or cost before the actual development of any software is the most crucial task of the present day software development project managers. Software project attributes are often measured in terms of linguistic values such as very low, low, Average, high and very high. The imprecise nature of such attributes constitutes uncertainty and vagueness in their subsequent interpretation. In this paper we propose a Fuzzy logic based model for software effort prediction. We feel that fuzzy Software cost estimation Model should be able to deal with imprecision and uncertainty associated with various parameter values. Fuzzy analogy model has been developed and validated upon student data.

Keywords : software cost estimation, effort prediction, fuzzy logic, linear regression.

GJCST-C Classification : D.2.8
Software Effort Prediction - A Fuzzy Logic Approach
Sanjay Kumar\textsuperscript{a}, Jaya Pal\textsuperscript{a} & Vandana Bhattacherjee\textsuperscript{p}

\textbf{Abstract} - Accuracy in the estimation of software Effort/Cost is one of the desirable criteria for any software cost estimation model. The estimation of effort or cost before the actual development of any software is the most crucial task of the present day software development project managers. Software project attributes are often measured in terms of linguistic values such as very low, low, Average, high and very high. The imprecise nature of such attributes constitutes uncertainty and vagueness in their subsequent interpretation. In this paper we propose a Fuzzy logic based model for software effort prediction. We feel that fuzzy Software cost estimation Model should be able to deal with imprecision and uncertainty associated with various parameter values. Fuzzy analogy model has been developed and validated upon student data.

\textbf{Keywords} : software cost estimation, effort prediction, fuzzy logic, linear regression.

\section{Introduction}

Accurate and timely prediction of the development effort and schedule required to develop a software system is one of the most critical activities in managing software projects. In addition software estimation has been identified as one of the three great challenges for half-century-old computer science. [19] In the last 30 years many different studies have been done in the area of Software Cost Estimation to improve the estimation accuracy and so many models are introduced. The rest of the paper contains the following sections as follows: section II represents Research Method, section III represents Experimental Results, and section IV represents Conclusion and Future Scope.

\subsection{Fuzzy Logic}

Intelligent Systems provide alternative paradigms aimed at facilitating the representation and manipulation of uncertain, incomplete, imprecise or noisy data. Fuzzy logic is a form of many-valued logic or probabilistic logic; it deals with reasoning that is approximate rather than fixed and exact.

The traditional approach to building system controllers requires a prior model of the system. The quality of the model, that is, loss of precision from linearization and/or uncertainties in the system's parameters negatively influences the quality of the resulting control. It is well known that the fuzzy theory not only provides natural tool for describing quantitative data but also generally produces good performance in many applications. In addition, fuzzy rules allow us to effectively classify data having non-axis-parallel decision boundaries, which is difficult for the conventional attribute-based methods. However, one of the difficulties with fuzzy decision trees is determining an appropriate set of membership functions representing fuzzy linguistic terms. Usually membership functions are given manually, however, it is difficult for even an expert to determine an appropriate set of membership functions when the volume and dimensionality of data are large.

At the same time, methods of soft computing such as fuzzy logic possess non-linear mapping capabilities, do not require an analytical model and can deal with uncertainties in the system's parameters. Although fuzzy logic deals with imprecise information, the information is processed in sound mathematical theory [40]. Based on the nature of fuzzy human thinking, Lofti Zadeh originated the "fuzzy logic" or "fuzzy set theory", in 1965. Fuzzy logic deals with the problems that have fuzziness or vagueness. In fuzzy set theory based on fuzzy logic a particular object has a degree of membership in a given set that may be anywhere in the range of 0 (completely not in the set) to 1 (completely in the set) [41].

For this reason fuzzy logic is often defined as multi-valued logic (0 to 1), compared to bi-valued Boolean logic [42]. Specifically, Fuzzy Logic offers a particularly convenient way to generate a keen mapping between input and output spaces thanks to fuzzy rules’ natural expression [2]. Fuzzy logic has been used in[36][37][38][39]. Fuzzy set theory and fuzzy logic are a highly suitable and applicable basis for developing knowledge-based systems This paper presents a fuzzy rule based system having two fuzzy inputs, namely Line of code (LOC) and Adjusted difficulty level (Adj. diff. level) and one output Development time (Devtime) as shown in Figure 1.

\textbf{Authors} \quad \textsuperscript{a} \textsuperscript{a} \textsuperscript{a} \textsuperscript{p} : Department of CSE, Birla Institute of Technology Ranchi, India. E-mails : k_sanjay71@yahoo.co.in, jayapal@bitmesra.ac.in, vbhattacharya@bitmesra.ac.in

© 2013 Global Journals Inc. (US)
Figure 1: Fuzzy Rule Based System

b) Fuzzy Rule Based System

A typical fuzzy logic system consists of four main components as shown in Figure.

- **Fuzzification**
  That contains predefined set of linguistic values. It converts non-fuzzy (Deterministic) inputs of fuzzy system into fuzzy inputs for inference mechanism.

- **Knowledge Base**
  That consists of two parts: database that defines linguistic variables (Fuzzy sets and rule base that represents the mapping of fuzzy input set into a fuzzy output set. Rules are fuzzy implications). Fuzzy sets, and rule base that represents the mapping of fuzzy input set into a fuzzy output set. Rules are fuzzy conditional statements (implications).

- **Decision Logic**
  That simulates human decision making based on fuzzy concepts. Conclusion of certain condition is derived by decision making logic.

- **Defuzzification**
  That converts rule base fuzzy outputs into non-fuzzy (.numerical) values.

Central mechanism of knowledge base and decision making logic considers the fuzzy extension of conventional rule inferencing concept to fuzzy rules inferencing. Premises and conclusions of rules now contain fuzzy values. These facts by definition describe practically continual input set of characteristics. In this manner, one rule can replace more conventional rules. Fuzzy inferencing rules generally connect m conditional variables X1,..., Xm to n consequent variables Y1,..., Yn in form of:

\[
\text{IF (X1 is A1 and \ldots \ldots \ldots Xm is Am) THEN (Y1 is B1 and \ldots \ldots \ldots Yn is Bn).}
\]

Where A1,..., Am and B1,..., Bn are linguistic terms of linguistic variables X1,...,Xm and Y1,..., Yn respectively.

The IF part is called the “antecedent” and the THEN part is called the “consequent”. To make a decision based on a set of rules, a rules-based system follows these steps:

1. All the rules that apply are invoked, using the membership functions and truth values obtained from the inputs (by a process called fuzzification), to determine the result of the antecedent.

2. This result in turn will be mapped into a membership function and truth value controlling the output variable. This process is known as implication. Two of the more common implication functions are: clipping (the fuzzy set is clipped to a value given by the level of activation of the input variables) and scaling (the fuzzy set is multiplied by a value given by the level of activation of the input variables).

3. These results are combined by a process called aggregation. One common approach for the aggregation involves using the “maximum” of the implicated sets.

4. Finally, a process known as defuzzification is used to compute a single value that is representative of the aggregated fuzzy set.

c) Multiple Regressions

A linear equation with three independent variables (multiple regressions) may be expressed as:

\[
y = b_0 + b_1 x_1 + b_2 x_2 \tag{1}
\]

Where b0, b1, and b2 are constants; x1, and x2 are the independent variables, and y is the dependent variable. The values of b0, b1, and b2 of the multiple regression. Equation may be obtained solving the following system of linear equations

\[
\Sigma y = nb_0 + b_1(\Sigma x_1) + b_2(\Sigma x_2) \tag{2}
\]

\[
\Sigma x_1 y = b_0(\Sigma x_1) + b_1(\Sigma x_1^2) + b_2(\Sigma x_1 x_2) \tag{3}
\]

\[
\Sigma x_2 y = b_0(\Sigma x_2) + b_1(\Sigma x_1 x_2) + b_2(\Sigma x_2^2) \tag{4}
\]

d) Evaluation Criteria

A common criterion for the evaluation of software effort models is the Magnitude of Relative Error (MRE) which is defined as follows:

\[
\text{MRE} = \frac{\text{Actual devtime–Predicted devtime}}{\text{Actual devtime}}
\]

The MRE value is calculated for each observation whose devtime is predicted. The
aggregation of MRE over multiple observations (N) can be achieved through the Mean MRE (MMRE) as follows:

$$\text{MMRE} = \frac{1}{N} \sum_{i=1}^{N} \text{MRE}$$

A complementary criterion is the prediction at level l, Pred (l) = k/N, where k is the number of observations where MRE is less than or equal to l, and N is the total number of observations. Thus, Pred (25) gives the percentage of development time of software which were predicted with a MRE less or equal than 0.25.

II. Research Method

a) Metrics Used

The following metrics have been used Line of Code (LOC), and Adjusted Difficulty Level (ADJ.DIFF.LEVEL) which is served as input to the Fuzzy Logic System. And one output Development Time (DEVTIME).

Description of Metrics

1. Line of Code (LOC): Loc is the total number of lines of code used to develop the software excluding the comment lines. This metric was measured on the scale of 0-60.
2. Adjusted Difficulty Level (ADJ.DIFF.LEVEL): This is the difficulty level of the programmers to develop the software which is further adjusted with the help of expert judgments. This metric was measured on the scale of 0-6.
3. Development Time (Devtime): It is the time taken to develop the software. This metric was measured on the scale of 0-24.

b) Data Gathered

The proposed model was validated by a data set collected from the BIT, students of MCA. This data set consists of 10 project data. The data set is applied to the proposed fuzzy model is shown in the Table 1.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Range</th>
<th>MF</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOC</td>
<td>0-60</td>
<td></td>
<td>a b c</td>
</tr>
<tr>
<td>ADJ.DIFF.LEVEL</td>
<td>0-6</td>
<td></td>
<td>a b c</td>
</tr>
<tr>
<td>DEVTIME</td>
<td>0-24</td>
<td></td>
<td>a b c</td>
</tr>
</tbody>
</table>

Output

The membership function plots corresponding to Table 2 are shown in figures2 (a), 2(b) and 2(c).

Figure 2(a) : LOC (Input)
d) Multiple Regressions
The same dataset has been used in Multiple Regression model to estimate the development time, which is to be used to develop a software. The estimated value of development time is compared with the actual value of development time, and with the help of this, using the evaluation criterion the MRE, MMRE and the Pred(0.25%) value is also calculated.

e) Evaluation criteria
For this model the same evaluation criterion is used. The criterion which is used to evaluate the fuzzy model.

III. Experimental Results
The results show more accuracy in case of effort estimation by the proposed fuzzy model. The result is shown in the Table 3, Table 4 and Table 5.

Table 3

<table>
<thead>
<tr>
<th>PROG_ID</th>
<th>DEVTIME (ACT)</th>
<th>DEVTIME (PRED) CALCULATED USING FUZZY LOGIC</th>
<th>MRE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>4</td>
<td>0.3333</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
<td>0.0000</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>10</td>
<td>0.0000</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>8</td>
<td>0.4667</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>4</td>
<td>0.3333</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>4</td>
<td>0.0000</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>4</td>
<td>0.0000</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>4</td>
<td>0.0000</td>
</tr>
<tr>
<td>9</td>
<td>7</td>
<td>4</td>
<td>0.4285</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>4</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

Table 4

<table>
<thead>
<tr>
<th>PROG_ID</th>
<th>DEVTIME (ACT)</th>
<th>DEVTIME (PRED) CALCULATED USING LINEAR REGRESSION</th>
<th>MRE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>4.2048</td>
<td>0.4016</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>5.7181</td>
<td>0.4295</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>9.8191</td>
<td>0.1808</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>13.7431</td>
<td>0.0838</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>4.2048</td>
<td>0.4016</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>2.0155</td>
<td>0.4961</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>2.7237</td>
<td>0.3191</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>3.3757</td>
<td>1.5606</td>
</tr>
<tr>
<td>9</td>
<td>7</td>
<td>6.2935</td>
<td>1.0092</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>6.9013</td>
<td>0.4753</td>
</tr>
</tbody>
</table>

Table 5: Prediction Results

<table>
<thead>
<tr>
<th>PROG_ID</th>
<th>Multiple Regression</th>
<th>Fuzzy Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min(MRE)</td>
<td>0.0838</td>
<td>0.0000</td>
</tr>
<tr>
<td>Max(MRE)</td>
<td>1.5606</td>
<td>0.4667</td>
</tr>
<tr>
<td>MMRE</td>
<td>0.5358</td>
<td>0.1762</td>
</tr>
<tr>
<td>Pred(25)</td>
<td>0.3</td>
<td>0.6</td>
</tr>
</tbody>
</table>

We have compared the actual development time with the predicted development time given by the model for each data set and found that difference between the actual devtime and predicted devtime. Then we calculated the MRE of each project and MMRE =0.1762 and pred(0.25%) which is 0.6. The same dataset has been tested using multiple regression model and the calculated MMR was used to further find out the MMRE=0.5358 and Pred(0.25%)=0.3. after going through the results we conclude that the proposed fuzzy model gives the better accuracy.

IV. Conclusion and Future Scope
The main benefit of this model is its good interpretability by using fuzzy rules and another great advantage of this research is that it can put together expert knowledge (Fuzzy rules) and project data into one general framework that may have a wide range of applicability in software estimation.

Further the comparison with multiple regression model to fuzzy logic model, and the results support the fuzzy logic model. In our future work we will test this model upon different real-time datasets. Datasets have been collected from the software engineering data repository [34]. These datasets have been used by various researchers in their work [4] [34] [35].

V. Acknowledgment
This research work has been partially funded by UGC [F.No.:33-61/2007(SR)] under financial grants for Major Research Project.
REFERENCES Références Referencias

10. Lionel C. Briand, “On the many ways Software Engineering can benefit from Knowledge Engineering”.


31. Internal Reports, Birla Institute of Technology Ranchi.


