Component Selection Efforts Estimation– a Fuzzy Logic Based Approach

Kirti Seth kirti.twins@gmail.com
Lecturer, CSE/GCET
UPTU
Greater Noida,India

Arun Sharma
Professor, AIIT arunsharma2303@gmail.com
Amity University,
Noida,India

Ashish Seth aseth@amity.edu
Lecturer, AIIT
Amity University
Noida, India

Abstract

Effort Estimation with good accuracy helps in managing overall budgeting and planning. The accuracy of these estimates is very less and most difficult to obtain, because no or very little detail about the project is known at the beginning. Due to architectural difference in CBS (Component Based Systems), this estimation becomes more crucial. Component-based development mainly involves the reuse of already developed components. The selection of best quality component is of prime concern for developing an overall quality product. CBS mainly involves two types of efforts: selection and integration. Present paper presents a fuzzy rule based model for estimating the efforts in selecting these Components for developing an application using CBSE approach.

Keywords: Component, Component Based software Engineering, Selection Efforts, Reusability, Fuzzy Rule Base.

1. INTRODUCTION

Commercial off the Shelf (COTS) software components or Component Based Software come from different source and have varied characteristics but by integrating them a software system can be formed [2]. Component Based Software Development (CBSD) approach is used widely in software industry. Efficient application can be developed through the integration of these components by using this approach. The components selected for the application are behind the success of these applications. Therefore a large amount of efforts has to be invested for selecting these components. There are several approaches for estimating such efforts. Present work proposes a fuzzy logic based approach for component selection. We identified the main factors which are useful in taking decision for selecting best suitable component. These factors are Reusability, Portability, Functionality, Security, and Performance. A rule base is prepared based on the effects of these on overall selection of component.

2. SELECTION EFFORTS

The main activities in any component based development require the following sequence of activities for the components.

Search -> Select -> Create/Adapt -> Integrate -> Maintain

Search and selection are the two most important activities, on which the quality of entire application depends. Therefore, sufficient efforts must be invested on selection of appropriate and better quality component. When a component is selected among various available components, a large amount of efforts is invested. These efforts should be estimated for better budgeting and planning of the software development. A lot of research work has been conducted by the researchers. Ali et al. [11] proposed the use of fuzzy sets in COCOMO 81 model [12]. They measured each cost driver of the intermediate COCOMO'81 model on a scale of six linguistic values ranging from very low, to nominal to extra high. For each cost driver and its associated linguistic values, they defined the corresponding fuzzy sets. These fuzzy sets are represented by trapezoidal shaped membership functions to finally estimate the overall cost. Musilek et al. [13] proposed F-COCOMO model, using fuzzy sets. F-COCOMO was based on fuzzy sets. Fuzzy sets may be applied to other models of cost estimation such as function point method. Mattson et al. [15] found that the software cost estimation model may be more user friendly by using concept of fuzzy sets. Valerie Maxville et al. [9] give a context driven component evaluation approach for estimating Component Selection Efforts. Mustafa Nouri et al. [14] estimate component selection by considering the NP-complete process of selecting a minimal set of components to satisfy a set of objectives. For this process, authors designed three variations of component selection and used approximation theory to find near optimal solution.

3. PROPOSED APPROACH

In this approach, following five factors that affect the Component Selection Efforts have been proposed.

3.1 Reusability

In case of Component Based Development, software reuse refers to the utilization of a software component with in an application, where the original motivation for constructing this component was other than for use in that application. In other words, reuse is the process of adapting a generalized component to various contexts of use. The idea of reusing software embodies several advantages. It improves productivity, maintainability and quality of software. Reusability is among one of the most basic properties of Component Based Development (CBD). In this model reusability is used as a factor. It is considered that if there are several components of the same type and among those available components the most appropriate component will be the one that has been reused so many times. So to select this component efforts invested will be low. Hence the selection efforts are indirectly proportional to the reusability.

3.2 Portability

This factor is defined as the ability of a component to be transferred from on environment to another with little modifications, if required. The component should be easily and quickly portable to specified new environments if and when necessary, with minimized porting costs and schedules. Therefore the specification of the component should be platform independent. Some components are platform independent that are highly portable. If a component is easily portable then the selection efforts are low.

3.3 Functionality

Functionality of a component depends upon the number of functions and their properties in these functions. It means that the component should provide the functions and services as per the requirement when used under the specified condition. Pre existing components with or minimum

changes will allow low cost, faster delivery of end product. If the functionality is high then the efforts invested are also high.

3.4 Security

The primary goals of software security are the preservation of the confidentiality, integrity, and availability of the information assets and resources that the software creates, stores, processes, or transmit, including the executing programs themselves. In other words, users of secure software have a reasonable expectation that their data is protected from unauthorized access or modification and that their data and applications remain available and stable. It refers how the component is able to control the unauthorized access to its provided services [5].If the component is highly secure then the efforts invested in selecting that component will be low.

3.5 Performance

This characteristic expresses the ability of a component to provide appropriate performance. This is affected by the component technology, mainly through resource usage by the run-time system but also by interaction mechanism. Component can be internally optimized to improve performance without affecting their specifications. Component should be tested on various platforms to check the performance. It means how well a component is providing the results. That is if all the parameters are given values then how accurately and fast it can produces the results [5]. Component Selection efforts are indirectly proportional to the performance.

4. FUZZY LOGIC

Fuzzy logic was proposed by Zadeh in 1965 [8] and since then it has been the subject of important investigations. It is a mathematical tool for dealing with uncertainty and also it provides a technique to deal with imprecision and information granularity. A keen mapping between input and output spaces may be developed with the help of fuzzy logic .Some major modules of fuzzy logic are as follows:

First stage transformed the classification tables into a continuous classification, this process is called fuzzification. These are then process in fuzzy domain by inference engine based on knowledge base (rule base and data base) supplied by domain experts. Finally the process of translating back fuzzy numbers into single “real world” values is named defuzzification.

5. PROPOSED FUZZY MODEL

There are five inputs to this fuzzy model, namely Reusability, Portability, Functionality, Security, and Performance. Figure 1 shows the fuzzy model.

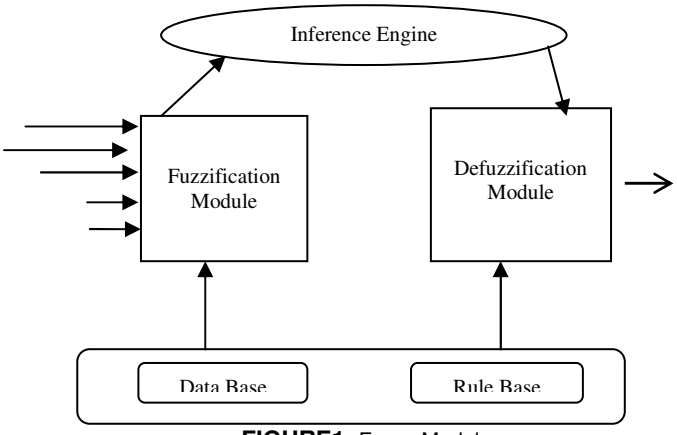
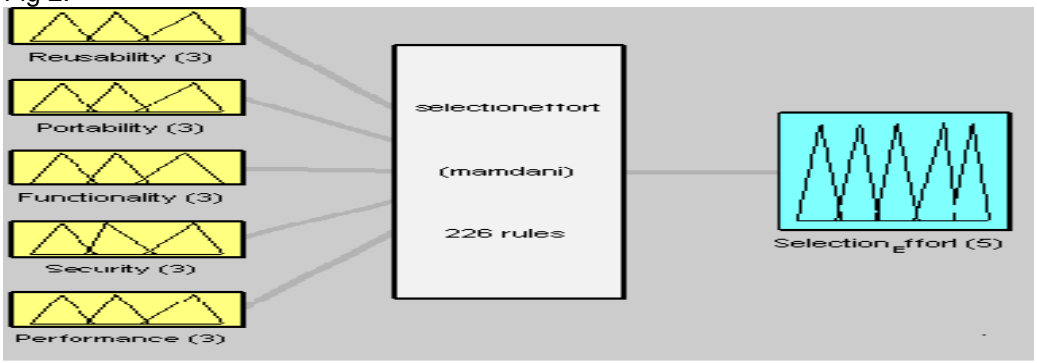


FIGURE1. Fuzzy Model.

This model considers all five inputs and provides a crisp value of Selection efforts using the Rule Base. All inputs can be classified into fuzzy sets viz. Low, Medium and High. The output Selection Efforts is classified as Very High, High, Medium, Low, and Very Low. In order to fuzzify the inputs, the following membership functions are chosen namely Low, Medium High. They are shown in Fig 2.



FIGUREZ Inputs and Outputs in our Fuzzy Model

Similarly the output variable i.e. Selection effort has five membership functions. All the inputs and outputs are fuzzified as shown in figure 2. All possible combination of inputs were considered which leads to 3^5 i.e. 243 sets. Selection Effort in case of all 243 combinations is classified as Very High, High, Medium, Low, very low by expert opinion. These lead to the Formation of 243 rules for the fuzzy model and some of them are shown below:

- If (Reusability is low) and (Portability is low) and (Functionality is low) and (Security is low) and (Performance is low) then Selection Efforts are very high. Selection Efforts are very high.
- If (Reusability is low) and (Portability is low) and (Functionality is low) and (Security is medium) and (Performance is low) then Selection Efforts are high.
- If (Reusability is low) and (Portability is low) and (Functionality is medium) and (Security is low) and (Performance is low) then Selection Efforts are very high formation of 243 rules for the fuzzy model and some of them are shown below:
- If (Reusability is low) and (Portability is low) and (Functionality is low) and (Security is low) and (Performance is low) then Selection Efforts are very high. Selection Efforts are very high.
- If (Reusability is low) and (Portability is low) and (Functionality is low) and (Security is medium) and (Performance is low) then Selection Efforts are high.
- If (Reusability is low) and (Portability is low) and (Functionality is medium) and (Security is low) and (Performance is low) then Selection Efforts are very high.

….
….

All 243 rules are entered and a Rule Base is created. a rule will be fired depending on a particular set of inputs. Mamdani style of inference is used.

System	Name='Selectioneffort', Type='mamdani', Version=2.0, NumInputs=5, NumOutputs=1, NumRules=243, AndMethod='min', OrMethod='max', ImpMethod='min', AggMethod='max', DefuzzMethod='centroid'
Input1	Name='Reusability', Range=[0 1], NumMFs=3, MF1='Low':'trimf',[0 0.16 0.33], MF2='medium':'trimf',[03 0.45 0.62], MF3='high':'trimf',[0.57 0.85 1]
Input2	Name='Portability', Range=[0 1], NumMFs=3, MF1='low':'trimf',[0 0.16 0.34], MF2='medium':'trimf',[0.30 0.45 0.62], MF3='high':'trimf',[0.56 0.85 1]

Input3	Name='Functionality', Range=[0 1], NumMFs=3, MF1='low':'trimf',[0 0.16 0.35], MF2='medium':'trimf',[0.30 0.45 0.62], MF3='high':'trimf',[0.56 0.80 1]
Input4	Name='Security', Range=[0 1], NumMFs=3, MF1='low':'trimf',[0 0.16 0.34], MF2='medium':'trimf',[0.3 0.40 0.65], MF3='high':'trimf',[0.60 0.85 1.0]
Input5	Name='Performance', Range=[0 1], NumMFs=3, MF1='low':'trimf',[0 0.16 0.33], MF2='medium':'trimf',[0.3 0.45 0.62], MF3='high':'trimf',[0.58 0.85 1.0]
Output1	Name='Selctioneffort', Range=[0 1], NumMFs=5, MF1='Very_Low':'trimf',[0 0.12 0.23], MF2='Low':'trimf',[0.2 0.32 0.42], MF3='Medium':'trimf',[0.40 .51 0.62], MF4='High':'trimf',[0.60 0.75 0.82], MF5='Very_High':'trimf',[0.80 .91 1.0]

Selection efforts are observed for a particular set of inputs using MATLAB Fuzzy tool box [8].

Table 1: Inputs and Outputs for Fuzzification

6. EVALUATION OF THE PROPOSED MODEL

We use the proposed model for five components and estimated the cost for each of these components. These selection efforts are on a scale of 0 to 1.

Project	Reusability	Portability	Functionality	Security	Performance	Selection efforts
P1	0.50	0.50	0.50	0.50	0.50	0.51
P2	0.17	0.45	0.23	0.62	0.18	0.472
P3	0.12	0.20	0.09	0.13	0.10	0.902
P4	0.72	0.90	0.75	0.85	0.97	0.311
P5	0.58	0.85	0.62	0.32	0.90	0.541

Table2: Results using rule Based system

From the table, it is clear that component P4 has the least selection effort i.e. will result in low cost development of the end product, while if the component P3 is to be used, the selection efforts were very high. So we can say that our model is able to predict the effort invested in selecting a component.

7. CONCLUSION:

For component-based development, efforts are mainly invested in selecting the appropriate component and then integrating it in the application. In this paper we have proposed a fuzzy rule based approach for estimating component selection efforts for these systems. The proposed approach is used to estimate efforts on some real time projects. However, the work further requires validation. For this purpose we are collecting more data from projects and by using Analytical Hierarchical approach we will validate our results.

8. REFERENCES

[1] Capers Jones, "Software Estimating Rules of Thumb", available at http://www.ieeexplore.ieee.org/iel1/2/10412/ 00485905.pdf.

[2] Richard E. Fairley "Recent Advances in Software Estimation Techniques", Management Associates Woodland Park, CO, USA 1992.

[3] I. Sommerville, "Software Engineering", Sixth Edition", Addison-Wesley Publishers Limited, 2001.

[4] IEEE standard Glossary of Software Engineering Technology, IEEE Std. 610 12-1990.

[5] A. Sharma, R. Kumar, P. S. Grover, "Empirical Evaluation and Validation of Complexity Metrics for Software Components", International Journal of Software Engineering and Knowledge Engineering, Vol. 18, Issue 7, 2008, pp: 919-931.
[6] P. S. Grover, R. Kumar, A. Sharma, "Few Useful Considerations for Maintaining Software Components and Component Based Systems", ACM SIGSOFT Software Engineering Notes, Vol. 32, Issue 5, 2007, pp: 1-5.
[7] Voas Jeffrey, Agresti, William W., Software Quality from a Behavioral Perspective IEEE Computer Society IT Pro, July- August 2004.
[8] Roger Jang and Ned Gulley, Fuzzy Logic Toolbox for MATLAB. User's Guide. The Math Works Inc, USA, 1995.
[9] Valerie Maxville, Jocelyn Armarego, Cheiou Peng Lam, "Intelligent Component Selection", proceedings of 28th annual international computer society and application conference.(COMPSAC '04).
[10] Carr R.D., Doddi, S., Konjevod, G., Marathe, and M.V.: On the red-blue set cover problem. In: SODA, 2000, pp. 345-353.
[11] Ali, I., A., Alain and K. Laila, "COCOMO cost model using Fuzzy Logic", 7th International Conference on Fuzzy Logic and Technology, Atlantic, New Jersey, March-2000.
[12] Bohem, B., "Software Engineering Economics", 1st Edition, Prentice-hall, Englewood Cliffs, New Jersey, 1981, ISBN: 0138221227.
[13] Musilek, P., W. Pedrycz, G. Succi and M. Reformat, "Software cost estimation with fuzzy models" ACM SIGAPP Applied Computer Review, Vol. 8, pp:24-29.
[14] Mostafa Nouri and Jafar Habibi, 2008, "Approximating Component Selection with General Costs"CSICC 2008, CCIS 6, pp: 61-68, .
[15] Matson, J.E., B.E. Barrett and J. M. Mellichamp, "Software Development Cost Estimation using Function Points", IEEE Transactions Software Engineering, Vol. 20, pp: 275-287.