Qualitative Analysis Model for Qualifying the Components for Reusability Using Fuzzy Approach

V. Subedha and S. Sridhar

Research Supervisor, Sathyabama University, Chennai, India
subedha@gmail.com, drssridhar@yahoo.com

Abstract. A reusable software component is a component in an executable form where it can be plugged into the environment. In order to reuse the high quality components from an existing environment, finding the measurable characteristics of the reusable software components and their corresponding metrics is an important issue. Although, some characteristics are impossible to measure or predict directly there are still many characteristics that can be used for measuring the reusable software component. We propose a qualitative analysis model to assess the quality of the component and this proposed model is feasible and can achieve high quality reusable components compare with other model in the existing literature. We measure the quality of the components for reuse with functional coverage report, software reuse metrics and minimum extraction time. Using these measures the identified set of components is classified into qualified set and not qualified set for reusability. The qualified set for reusability will give high potential and high quality reusable components which will increase the reuse frequency and reuse utility level. The reusability degree of the component can be obtained with the help of Fuzzy Rules.

Keywords: Software reuse, reusability metrics, quality of reusability, fuzzy logic, component assessment model.

1 Introduction

Software reuse [1] is the process of using the existing software assets. Software assets or components include all software products, schemas, architectures, requirements, proposals, specifications, design, user manuals, plans, interface, data, templates and test suites. The reuse experts pay much attention to extract the component but neglect the quality of the components to be reused. To achieve a significant reuse with components, there should be a well define quality model for evaluating the reusability [2].

If there is more number of components available in the repositories, it is necessary to discover some software reuse metrics to qualify the components based on characteristics of components [3]. Individual software metric cannot measure the overall quality characteristics of the reusable software component. So, in this paper we propose a quality analysis model for assessing the quality of the reusable software components. The fundamental motivation of this proposed model is to achieve high

V.V. Das and P.M. El-Kafrawy (Eds.): SPIT 2012, LNICST 117, pp. 160–167, 2014.

potential benefits of reuse. Three important characteristics of reusability are i) Functional usefulness of the component which is to be reused ii) Time for Extracting the Component from the existing library and iii) Quality of the component

So, we combine the metrics based on the functional coverage report, software reuse metrics and minimum extraction time and we use fuzzy rules for qualifying the component for reusability. The rest of the paper is structured as follows: section 2 briefly describes the existing literature. In section 3, we describe the proposed qualitative analysis model for qualification. Section 4 gives the definitions of measures and metrics. In section 5, the process of qualitative analysis model for the qualification phase using Fuzzy rules. Section 6 presents the evaluation of qualitative analysis model, while section 7 concludes this work.

2 Related Work

Several works has been done in this area. Some of them are as:

In 2005, Richard W. Selby [4], investigated, analyzed, and evaluated software reusability by mining software repositories from a NASA software development environment that actively reuses software. The author achieved an average reuse of 32 percent per project. In 2006, Parvinder S. Sandhu and Hardeep Singh [5] proposed reusability evaluation model for assessing the reusability of software components. The authors proposed Neuro-fuzzy Inference engine can be used to evaluate the reusability. In 2007, Parvinder S. Sandhu Pavel Blecharz and Hardeep Singh [6] proposed Quantitative Investigation of impact of the factors contribution towards measuring the reusability of software components which helps to evaluate the quality of the components. They used Taguchi approach in analyzing the significance of different attributes in deciding the reusability level of a particular component.

In 2008, GUI GUI and Paul D. Scott[7] proposed new measure of coupling and cohesion to assess the reusability of components. They shown that the new measures proposed by them was consistently superior at the time of measuring the component reusability. They used five metrics for coupling and five metrics for cohesion and they were very good predictors for evaluation the reusability of the component.

In 2009, Parvinder S. Sandhu, Harpreet Kaur and Amanpreet Singh [8] proposed reusability evaluation system for object oriented software components. In 2010, Sonia Manhas, Rajeev Vashisht, Parvinder S. Sandhu and Nirvair Neeru [9] proposed reusability evaluation model for assessing reusability of software components. The different neural network approaches are used for the modeling of the reusability data.

In 2011, Fazal-e-Amin, Ahmad Kamil Mahmood and Alan Oxley [10] proposed reusability attribute model for assessing reusability of software components. The proposed model is derived using the GQM approach. In 2012, Ajay Kumar [12] proposed a model for classification of the reusability of software components using support vector machine. In all existing model the static metrics are defined to evaluate the quality of the components for reusability where we propose dynamic metrics to evaluate the quality of the components for reusability.

3 Proposed Qualitative Analysis Model for Qualification

The proposed qualitative analysis model is used to assess the r component quality for high potential benefits and high quality reuse in order to increase the productivity. The specific steps involved in the qualitative analysis model are

1. Identify the characteristics for quality analysis of reusable software components
2. Selection of metrics to measure the identified characteristics and this metrics will serves as a qualifier for quality analysis in the qualification phase
3. Integrate all the metrics and give the priority to the metrics
4. In measurement invocation the assessment was carried out using the qualifiers and the entire set of identified reusable components is classified as qualified set for reuse and not qualified set for reuse.

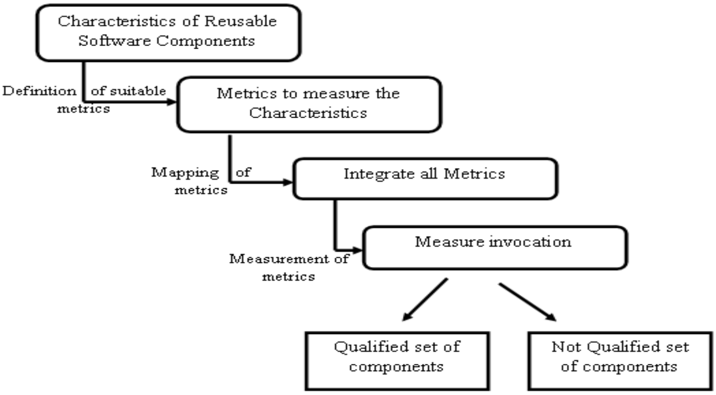
Characteristics of Reusable Software Components
Definition of suitable metrics
Metrics to measure the Characteristics
Mapping of metrics
Integrate all Metrics
Measurement of metrics
Measure invocation
Qualified set of components
Not Qualified set of components

Fig. 1. Qualitative Analysis Model

As shown in the fig. 1. The first step is to discover the characteristics of reusable software components which can be applicable to the environment.

The second step is to select the metrics to measure the characteristics and derive the quality attributes and also this metrics will serve as a qualifier for quality analysis in the qualification phase. This is done by collecting the list of metric and choosing the metrics which is easy, robust and useful for the software component reusability. Once we identified the qualifier in next step, we have to define the computing values of the metrics and integrate them. In the fourth step invoke the measurement and this step classifies the entire identified set as qualified or not qualified for reusability.

4 Measures and Metrics in Qualitative Analysis Model

In software engineering measurement, a metric is a quantitative indicator of a software attribute. A metrics model specifies relationships between metrics and the characteristics being measured by these metrics. In this section a description of the measures and metrics which are used to assess reusability is provided.

4.1 Functional Verification Measure

The first step involved in functional verification of the component in collecting the requirements of the reusability. By using coverage directed test generation for functional verification by genetic algorithm the reuser generates, executes a set of test cases and functional coverage report of the component is collected. We use the statement coverage and branch coverage as metrics for coverage analysis to qualify the candidate component for reusability.

4.2 Reuse Measure

The functional usefulness of the component is measured by reuse frequency which is an indirect measure of the functional usefulness of a component. We measure the functional usefulness as that frequently used system is a good candidate for reuse in context level in similar domain. Hence we choose the metrics reuse frequency as a qualifier for qualifying the components. Reuse frequency of each component can be calculated using the equation (1).

$$\text{Reuse Frequency} = \frac{n(C)}{\frac{1}{n}\sum_{i=1}^{n} n(Si)} \tag{1}$$

where n(C) is total number of reference to the Component, n(Si) is total number of reference for each Standard Components in the existing environment & n is the total number of component in the existing environment.

4.3 Time Measure

The components having the extraction time less than the average extraction time is qualified for reuse. The reason for choosing the extraction time as metrics is to speed up the process of reuse. The extraction time and the optimal path for the extraction are calculated using a scheme called as minimum Extraction Time First.

5 Process of Qualitative Analysis Model Using Fuzzy Rules

Each individual metrics plays an important role to measure the individual characteristic of the reusable components. But these metrics individually cannot provide an overall measurement, so we combine these metrics as the measurement of the reusable software component. In this model we assign different weights to different metrics and it is combining into overall measurement and the fuzzy rules are used to classify the components for reusable or not for reusable.

5.1 Metrics Suite Combination for Higher-Level

The metric data for the component qualification were collected using the different tools and they are combined to higher level to measure the quality. The four set of primitive metrics are combine into one higher-level measurement as follows:

LV_1: Linguistic Values for Statement coverage metric & W_1: Weight for LV_1
LV_2: Linguistic Values for Branch coverage metric & W_2: Weight for LV_2
LV_3: Linguistic Values for Extraction time metric & W_3: Weight for LV_3
LV_4: Linguistic Values for reuse frequency metric & W_4: Weight for LV_4
We combine these four metrics into a quality measurement as formula (2)

$$QM = LV_i\, W_i \quad \text{Where} \quad \sum_{i=1}^{4} wi = 1 \tag{2}$$

5.2 Design of Linguistic Variables for Selected Metrics

Linguistic variables are then assigned to the metrics based on their values. The assignment of the linguistic variable depends on the range of the metrics measurement. We recommend that all linguistic value of each primitive metric shall be normalized between LOW, MEDIUM and HIGH

Linguistic variables for statement coverage: Statement coverage is assigned with two linguistic variables LOW and HIGH based on the coverage report. If the statement coverage is 100% then the linguistic variable is HIGH else it is LOW.

Linguistic variables for Branch coverage: Branch coverage is assigned with three linguistic variables LOW, MEDIUM and HIGH based on the coverage report. If the branch coverage is >90% then the linguistic variable is HIGH else if it is within the range 85% - 90% then it is MEDIUM else it is LOW.

Linguistic variables for Extraction time: The components having the extraction time less than the average extraction time is qualified for reuse from the component repositories. Extraction time is assigned with two linguistic variables as LOW and HIGH. If the extraction time is less than the average extraction time is assigned with linguistic variable HIGH else it is assigned with LOW.

Linguistic variables for reuse frequency: Reuse-Frequency is assigned with two linguistic variables LOW and HIGH as constants in the range of less than 1 and greater than 1.

Linguistic variables for reusability degree: Reusability degree is the output parameter which is use to measure the reusability and assigned with two linguistic variables LOW and HIGH as constants in the range of less than or equal to 0.8 and greater than 0.8.

5.3 Fuzzy Rules for Qualifying the Components

Fuzzy inference is the process of formulating the mapping from a given input to an output using fuzzy logic. The mapping then provides a basis for decision-making. The process of fuzzy inference involves all of the pieces like membership functions, fuzzy logic operators and if-then rules. Consider a fuzzy system having the following rule base.
Rule 1 : if u is A1 and x is B1 and y is C1 and z is D1, then R=w11+w21+w31+w41
Rule 2 : if u is A2 and x is B2 and y is C2 and z is D2, then R=w12+w22+w32+w42.
. . . .
Rule n : if u is An and x is Bn and y is Cn and z is Dn, then R= w1n+w2n+w3n+w4n
Let the membership functions of fuzzy sets Ai, Bi, Ci, Di where i=1, 2, 3…..n be μ_{Ai}, μ_{Bi}, μ_{Ci}, μ_{Di}. Evaluating the rule premises results in (3)

$$R_i = \mu_{Ai}(u) + \mu_{Bi}(x) + \mu_{Ci}(y) + \mu_{Di}(z) \tag{3}$$

where R_i is the reusability degree of the i^{th} component.

With the help of the fuzzy-rules the knowledge base for calculating the reusability degree of the reusable software components is calculated. As there are totally four inputs, in which three inputs have two membership functions and other input have three membership functions. After all the possible combinations the size of the rule base comes out to be 2*3*2*2=24. Examples rules could be :
If (statement coverage is LOW) and (branch coverage is LOW) and (extraction time is LOW) and (reuse frequency is LOW) then (Reusability Degree is LOW)
If (statement coverage is HIGH) and (branch coverage is HIGH) and (extraction time is HIGH) and (reuse frequency is LOW) then (Reusability Degree is HIGH)

5.4 Weight Value Selection

After collecting the metrics for assessing the reusability degree and assigning the linguistic variable for each metrics, we select the suitable weight values to compute the quality measurement value for assessing the reusability degree. Production rules can be generated to select the weight value.
R11: if LV_1 = LOW then w11 R12: if LV1 = HIGH then w12 R21: if LV_2 = LOW then w21 R22: if LV2 = MEDIUM then w22 R23: if LV2 = HIGH then w23
R31: if LV3 = LOW then w31 R32: if LV3 = HIGH then w32
R41: if LV4 = LOW then w41 R42: if LV4 = HIGH then w42

6 Evaluation of Qualitative Analysis Model

In this section we describe experiments with proposed qualitative analysis model qualifying the components with our own test cases. All necessary information for calculating the quality of the components is given in the following Table 1. Accordingly, Table 2 presents the measurement data for high reuse components and it shows that in general 40 to 50 percent of identified components for possible reuse.

Table 1. Identified Set of components for reusability with all metrics

Component Number	Statement Coverage	Branch Coverage	Extraction Time	Reuse Frequency
1	95%	85%	0.941	0.80
2	100%	95%	0.047	1.70
3	100%	80%	1.023	0.71
4	100%	95%	0.177	0.98
5	100%	90%	0.056	0.41
6	100%	85%	0.320	1.50
7	100%	95%	0.913	1.40
8	100%	95%	0.781	0.30
9	90%	75%	1.001	0.79
10	100%	80%	0.328	1.25

Table 2. Classified Set of components for reuse and not reuse

Component Number	Weight for reusability degree using fuzzy rules	Linguistic variables For Reusability degree	Status of the Component
1	0.36	LOW	Not Qualified
2	1.0	HIGH	Qualified
3	0.43	LOW	Not Qualified
4	0.84	HIGH	Qualified
5	0.80	LOW	Not Qualified
6	0.91	HIGH	Qualified
7	0.85	HIGH	Qualified
8	0.69	LOW	Not Qualified
9	0.16	LOW	Not Qualified
10	0.74	LOW	Not Qualified

The Component reuse percentage for reusability is calculated and shown in the following figure 2.

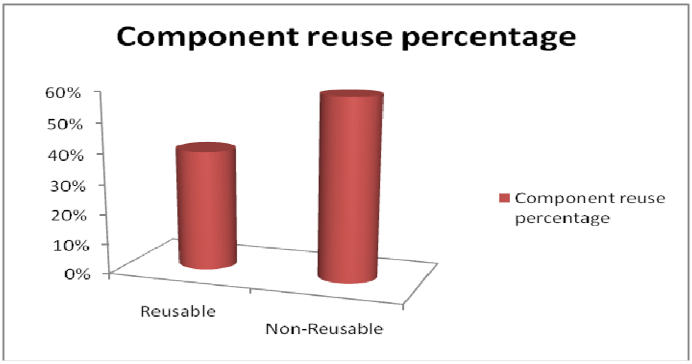

Fig. 2. Bar chart for component reuses percentage

7 Conclusion

One of the main important issue in reusing the software component is to optimize the process of identifying the suitable components for a given requirements. Effective reuse is only possible with effective assessment and classification. In this paper we have presented the stat-of-art of the qualitative analysis model especially in the context of software reuse.

We proposed a new model for quality assessment based on fuzzy rules. The main purpose of this model is to measure the quality of the component in order to realize the reusability of component effectively and to identify the best components in terms of their reusability. We need to expand the number of metrics in order to provide adequate coverage with respect to reusability of a component. We intend to apply this model in a wider variety of applications through several platforms to determine the impact and to design an automated tool for component classification.

References

1. Frakes, W.B., Kang, K.: Software Reuse Research: Status and Future. IEEE Transactions on Software Engineering 31(7), 529–536 (2005)
2. Choi, S.W., Her, J.S., Kim, S.D.: Qos Metrics for Evaluating Services from the Perspective of Service Provides. In: Proceedings of IEEE International Conference on e-business engineering (ICEBE 2007), pp. 622–625 (2007)
3. Singh, S., Thapa, M., Singh, S., Singh, G.: Software Engineering – Survey of Reusability Based on Software Component. International Journal of Computer Applications (0975 – 8887) 8(12), 39–42 (2010)
4. Selby, R.W.: Enabling Reuse-Based Software Development of Large-Scale Systems. IEEE Transaction of Software Engineering 31(6), 495–510 (2005)
5. Sandhu, S., Singh, H.: Automatic Reusability Appraisal of Software Components using Neuro-Fuzzy Approach. International Journal of Information Technology 3(3), 209–214 (2006)
6. Sandhu, P.S., Blecharz, P., Singh, H.: A Taguchi Approach to Investigate Impact of Factors for Reusability of Software Components. World Academy of Science, Engineering and Technology, 135–140 (September 2007)
7. Gui, G., Scott, P.D.: New coupling and cohesion Metrics for Evaluation of Software Component Reusability. In: Proc. ICYCS, pp. 1181–1186 (2008)
8. Sandhu, P.S., Kaur, H., Singh, A.: Modeling of Reusability of Object Oriented Software System. World Academy of Science, Engineering and Technology (30), 162–165 (2009)
9. Manhas, S., Vashisht, R., Sandhu, P.S., Neeru, N.: Reusability Evaluation Model for Procedure Based Software Systems. International Journal of Computer and Electrical Engineering 2(6), 1107–1110 (2010)
10. Fazal-e-Amin, Mahmood, A.K., Oxley, A.: Reusability Assessment of Open Source Components for Software Product Lines. International Journal on New Computer Architectures and Their Applications (IJNCAA) 1(3), 519–533 (2011)
11. Kumar, A.: Measuring Software Reusability using SVM based Classifier Approach. International Journal of Information Technology and Knowledge Management 5(1), 205–209 (2012)