Success Evaluation of Software Reuse Factors at the Development Stages of Reuse Process Model

Anjali Rai (student, RIEIT)  
Computer science engineering  
RIEIT  
Ropar, India  
anjalirai534@gmail.com

Harish kundra (HOD, RIEIT)  
Computer Science Engineering  
RIEIT  
Ropar, India

Pooja Rani (lecturer, RIEIT)  
Computer Science Engineering  
RIEIT  
Ropar, India  
poojashrm27@gmail.com

Abstract- A great deal of research over the past several years proves that clustering of components is must for the evaluation of success and failure but, early clustering was not applied at early stages as soon as the need for developing a new system using software reusability. In later approach evaluation was applied to few matrices or with functional matrices mainly but for more accurate evaluation we have to apply clustering from starting stage of development .So here we are applying two different and efficient clustering algorithms that are hierarchical clustering and K-mean clustering at two different stages which are development phases of software reuse process model. Data set consist of factors which are necessary for the development of the software system and thus result comes up with more accuracy of success and failure of software reusability.

Keywords- Reuse process model, Reuse, hierarchical clustering, k-mean clustering

I. INTRODUCTION

In Software Engineering, Component Based Software Engineering (CBSE) is concerned with the assembly of pre-existing software components (reusable components) that leads to a software system that responds to client-specific requirements. Component-Based Software Engineering (CBSE) is concerned with the development of software systems from reusable parts (components), the development of components, and system maintenance and improvement by means of component replacement or customization. Reusable Software is already existed software which is used for developing the new software products, with less effort, time and cost. Software reuse has a positive impact on software quality, as well as on software costs, and productivity. According to Krueger’s general view of software reuse [Kru92]: “Software reuse is the process of creating software systems from existing software rather than building them from scratch”.

A. Reuse process model

The given model (fig. 1) is based on “Software development for reuse process model” which can be combined with any of the software life cycle models. This model depicts how the stored component will be, select for developing the desired application, compatibility with another components during the software development. Here we can see the different parts of this model which are used for software development using reusable components.

![Fig-1](image-url)  
Enhanced Software development for reuse process model

In this model two stages are defined and thus success factors are taken at each stage .
1) Domain Analysis- Operating system factors included at this stage are following:-

* Load Balancing – The system should implement an intelligent process placement mechanism. Expressed in (yes, no)
- Binary Compatibility – The system should provide the transparent remote execution, and load balancing and any other features, without requiring modifications to an application or relinking. Expressed in (yes, no)

- High Availability – This was been discussed many times and simple means that when a node fails or is removed from the cluster, the system should continue to operate. Expressed in (yes, no)

- Portable – The system should be readily portable to other architectures. Expressed in (yes, no)

- Memory management – It tells the availability of memory for the current storage and execution of the program. Expressed in numeric.

Domain analysis means to identify the requirements for the different system to be developed; here we gave the model for multiple system requirements. For example to develop an operating system to add date and time component we will use the preexisting code for date and time.

Type of clustering used here is hierarchical clustering. Following is the flow chart of the algorithm.

![Flow chart of hierarchical clustering algorithm](image)

Fig-2
Flow chart of hierarchical clustering algorithm

2) Architectural Design.- During architectural design extra effort is now spent to make components more general and more reusable. These components are then incorporated into the storage together with any information that might be of help at a later stage, when searching for or retrieving components. In this phase all the functional requirements are gathered.

Here Design factors of components are evaluated, following data set is used:-

- Modularity–Modern software engineering and programming practices stress **modularity**, especially at the software architecture level. Expressed in (yes, no)

- Grouping–The idea is to group logically related parts (data and operations) to ease construction and maintenance. Key concept here is functional abstraction. Expressed in (yes, no)

- Platform independence—is crucial to enable software to be used on hardware platforms and in software environments other than the ones on which it has been developed. Expressed in (yes, no)

- Use case–How many use cases have been used. Expressed in numeric

- Information hiding–concerns the separation of concerns. The underlying idea is to make those parts of a code invisible that are of no concern to the user or environment, making the total software package a set or hierarchy of black boxes, each performing a specified function. Key concepts here are data abstraction and specification. Expressed in (yes, no)

- Components–Number of components to be used. Expressed in numeric.

Type of clustering used at this stage is k-mean clustering algorithm. Following is the flow chart of the algorithm.

![Flow chart of k-mean clustering algorithm](image)

Fig-3
Flow chart of k-mean clustering algorithm

II. PROBLEM FORMULATION

Problem with early research is that clustering of components was must for the evaluation of success and failure but, clustering was not applied at early stages as soon as the need for developing a new system using software reusability was required and thus evaluation of success factors was delayed later on which does not provide that much accuracy, which is must for the use of software reusability.

In later approach evaluation was applied to few matrices or with functional matrices mainly but for more accurate evaluation we have to apply clustering from starting stage of development.

So here we are applying two different and efficient clustering algorithms at two different stages which are development phases of software reuse process model.
Which will give more accuracy of success and failure of software reusability.

III. PROPOSED METHODOLOGY
The success of software reuse can be measured by following steps.

A. Selection of Dataset and Factors at first stage of software reuse model that is domain analysis stage
The datasets are generated from survey, interviews and questionnaires with the organization, related to the software to be developed. For example to develop an operating system following factors are considered:
- Load Balancing
- Binary Compatibility
- High Availability
- Portability
- Memory management

B. Collect or create the relevant data
Collect the relevant data from the dataset, which are required for the success of software reuse.

C. Perform clustering
The Clustering is an approach that uses software measurement data for analyzing software quality. So here hierarchical clustering is performed.

D. Comparison
The comparisons are made on the basis of the least value of Accuracy, Precision, and Recall values. In case of the twocluster based problem, the confusion matrix has four categories: True positives (TP) are modules correctly classified as faulty modules. False positives (FP) refer to faultfree modules incorrectly labelled as faulty modules. True negatives (TN) correspond to fault-free modules correctly classified as such. Finally, false negatives (FN) refer to faulty modules incorrectly classified as fault-free modules as shown in fig 3.

<table>
<thead>
<tr>
<th>Predicted Project</th>
<th>Real Data Value of Project Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Success</td>
<td>TP</td>
</tr>
<tr>
<td>Failure</td>
<td>FP</td>
</tr>
<tr>
<td>Success</td>
<td>TP</td>
</tr>
<tr>
<td>Failure</td>
<td>FN</td>
</tr>
</tbody>
</table>

Fig 3
Confusion metrics of predictions outcomes

E. Selection of Dataset and Factors at second stage of software reuse model that is architectural design stage
The datasets are generated from survey, interviews and questionnaires with the organization, related to the software to be developed. During architectural design extra effort is now spent to make components more general and more reusable. Thus following design factors are considered:
- Use case
- Information hiding
- No. of Components

F. Perform clustering:
The Clustering is an approach that uses software measurement data for analyzing software quality. So here k-mean clustering is performed.

G. Comparison
The comparisons are made on the basis of the least value of Accuracy, Precision, and Recall values. In case of the twocluster based problem, the confusion matrix has four categories: True positives (TP) are modules correctly classified as faulty modules. False positives (FP) refer to faultfree modules incorrectly labelled as faulty modules. True negatives (TN) correspond to fault-free modules correctly classified as such. Finally, false negatives (FN) refer to faulty modules incorrectly classified as fault-free modules as shown in fig 3.

H. Conclusion
The conclusions are made on the basis of the comparison made in the previous section.

IV. RESULT AND DISCUSSION
The implementation of the purposed work is done in open source tool known as WEKA 3.7
At first stage-:
First the dataset is loaded in WEKA environment. The metadata view of the input dataset is shown in the figure 2.
The dataset includes all the factors or variables that are considered in the reusability of software. Here operating system factors are taken. Following is the view of dataset.

Parameters of the hierarchical algorithm is shown in following

The Text view of Cluster Assignment is shown in Fig.6. The figure shows that the 14 examples are assigned to cluster 0, 2 examples are assigned to cluster 1.

At second stage:-
First the dataset is loaded in WEKA environment. The metadata view of the input dataset is shown in the figure 2. The dataset includes all the factors or variables that are considered in the reusability of software. Here design factors of system are taken. Following is the view of dataset.
K-mean clustering is applied to the dataset and following parameters are taken for k-mean clustering.

The Text view of Cluster Assignment is shown in Fig.9. The figure shows that the 11 examples are assigned to cluster 0, 3 examples are assigned to cluster 1.

V. CONCLUSION

Our conclusion comes up with some benefits of purposed approach over early approach. Reuse based approaches emphasize cost reduction as a means of increasing productivity. From an accounting perspective there are different ways of achieving this. One way is the amortization of the development and maintenance cost of assets over multiple projects. Another way is the avoidance of cost in later projects through the use of results of earlier projects. Here we simply reached to a conclusion that by using two different approaches of clustering and by then evaluating their success it becomes more clear and efficient about the success and failure of using reusable approach to develop the software. As we came with the improved results than the early approaches as the success is better than early approach.

REFERENCES

