A k-means Clustering Based Approach for Evaluation of Success of Software Reuse

Jagmeet Kaur, Surbhi Gupta, Sheetal Kundra

Abstract—A great deal of research over the past several years has been devoted to the development of methodologies to create reusable software components and component libraries. But the issue of how to find the contribution of the factor towards the successfulness of the reuse program is still in the naïve stage and very less work is done on the modeling of the success of the reuse. The success and failure factors are the key factors that predict the successful reuse of software. An algorithm has been proposed in which the inputs can be given to k-Means Clustering system in form of tuned values of the Data Factors and the developed model shows the high precision results, which describe the success of software reuse.

Keywords—k-means, Reuse and Machine learning.

I. INTRODUCTION

SOFTWARE Reuse and Success factors: - Systematic reuse is generally recognized as a key technology for improving software productivity and quality (Mili et al. 1995), possibly with a higher payoff than process improvement or process automation(Boehm 1993). Software reuse is the process whereby an organization defines a set of systematic operating procedures to specify, produce, classify, retrieve, and adapt software artefacts for the purpose of using them in its development activities. In the April 2002 TSE article Success and Failure Factors in Software Reuse [1], Morisio et.al. sought key factors that predicted for successful software reuse. Their data came from a set of structured interviews conducted with project managers of 24 European projects from 19 companies in the period 1994 to 1997. Those projects were trying to achieve company-wide reuse of between one to a hundred assets. Nine of those 24 projects were judged by their respective managers as failures. Morisio et.al. employed a well-designed interview process to collect a wide set of project attributes (for a complete listing of those attributes, see the appendix). There is much that is exemplary in the approach taken by Morisio et.al. For example, their data collection method is well-documented.

Three main causes of failure were not introducing reuse-specific processes, not modifying non-reuse processes, and not considering human factors. The root cause was a lack of commitment by top management, or non-awareness of the importance of those factors, often coupled with the belief that using the object-oriented approach or setting up a repository seamlessly is all that is necessary to achieve success in reuse.

Conversely, successes were achieved when, given a potential for reuse because of commonality among applications, management committed to introducing reuse processes, modifying non-reuse processes, and addressing human factors.

While addressing those three issues turned out to be essential, the lower-level details of how to address them varied greatly: for instance, companies produced large-grained or small-grained reusable assets, did or did not perform domain analysis, did or did not use dedicated reuse groups, used specific tools for the repository or no tools. As far as these choices are concerned, the key point seems to be the sustainability of the approach and its suitability to the context of the company.

There are three types of Factors that are considered here as:
- High Level Control variables
- State Variables
- Low Level Control Variables

A. High Level Control Variable

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Response for Reuse</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top Management commitment</td>
<td>Yes</td>
</tr>
<tr>
<td>Key Reuse Roles</td>
<td>Yes</td>
</tr>
<tr>
<td>Reuse Process</td>
<td>Yes</td>
</tr>
<tr>
<td>Non Reuse Process Modified</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Jagmeet Kaur is M.Tech. CSE student at RIEIT, Rail Majra, Punjab, India
Ms. Sheetal Kundra is Sr. Lecturer at Dept. of CSE, RIEIT, Rail Majra, Punjab, India.
Ms. Surbhi Gupta is Senior Lecturer, Department of Computer Science & Engineering, Rayat Institute of Engineering & Information Technology, Punjab, India.
Human Factors | Yes | No
--- | --- | ---
Repository of assets | Yes | No

Note that all 23 projects seen in this data set used a repository; i.e. this data set could never be used to refute claims that a repository is useless. Nevertheless, like Morisio et al., we believe that reuse products have to be kept in some sort of repository to enable reuse.

B. State Variables:

The state Variables, these are the attributes over which a company has no control.

<table>
<thead>
<tr>
<th>Attributes</th>
<th>No. Of Staff</th>
<th>Overall Staff</th>
<th>Type of Production</th>
<th>Product type</th>
<th>SP Maturity Level</th>
<th>Application Domain</th>
<th>Size of Baseline</th>
<th>Staff Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Large</td>
<td>Medium</td>
<td>Small</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Large</td>
<td>Medium</td>
<td>Small</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Product-family</td>
<td>Project related</td>
<td>Isolated</td>
<td>Embedded</td>
<td>CMM Level 3</td>
<td>TLC</td>
<td>Large</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Small</td>
<td>Low</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

C. Low Level Control Variables

These are the specific approaches to the implementation of reuse.

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Reuse Approach</th>
<th>Domain Analysis</th>
<th>Configuration Management</th>
<th>Rewards Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Loosly Coupled</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Tightly Coupled</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

D. k-means Clustering Algorithm

Clustering (or cluster analysis) aims to organize a collection of data items into clusters, such that items within a cluster are more “similar” to each other than they are to items in the other clusters. This notion of similarity can be expressed in very different ways, according to the purpose of the study, to domain-specific assumptions and to prior knowledge of the problem. Clustering is usually performed when no information is available concerning the membership of data items to predefined classes.

![Fig.1 Flowchart of k-means Algorithm.](image)

II. PROBLEM FORMULATION

A great deal of research over the past several years has been devoted to the development of methodologies to create reusable software components and component libraries. But the issue of how to find the contribution of the factor towards the successfullness of the reuse program is still in the naïve stage and very less work is done on the modeling of the success if the reuse. Our approach, for evaluation success of software reuse, is based on software models and metrics. As the exact relationship between the attributes of the reuse success is difficult to establish so a Clustering based approach could serve as an economical, automatic tool to generate ranking of reuse success by formulating the relationship based on its training.

III. PROPOSED METHODOLOGY

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

A. Selection of Dataset and Factors

The datasets are generated from the interviews and questionnaires with the organization, related to the software to be developed. There are three types of factors that are considered are:
a) High level control variables  
b) State variables  
c) Low-level control variables.

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. In this step, \textit{k}-Means clustering algorithm is used for partitioning the data into different levels of reusability value based on the structural metric values as \textit{k-means} is the well known approach that classify data into different \textit{k} groups where \textit{K} is a positive integer, based on the attributes or some features. Grouping of data is done on the basis of minimizing sum of squares of distances between data and their cluster centroid.

D. Comparison
The comparisons are made on the basis of the least value of Accuracy, Precision, and Recall values. In case of the two-cluster based problem, the confusion matrix has four categories: True positives (TP) are modules correctly classified as faulty modules. False positives (FP) refer to fault-free 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 Table 1.

<table>
<thead>
<tr>
<th>Table IV</th>
<th>Confusion Matrix of Prediction Outcomes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicted Project</td>
<td>Real Data Value of Project Status</td>
</tr>
<tr>
<td>Success</td>
<td>Failure</td>
</tr>
<tr>
<td>Success</td>
<td>TP</td>
</tr>
<tr>
<td>Failure</td>
<td>FN</td>
</tr>
</tbody>
</table>

With help of the confusion matrix values the precision and recall values are calculated described below:

\textbf{Precision:}
Precision for a class is the number of true positives (i.e. the number of items correctly labeled as belonging to the positive class) divided by the total number of elements labeled as belonging to the positive class (i.e. the sum of true positives and false positives, which are items incorrectly labeled as belonging to the class). The equation is:

\[ \text{Precision} = \frac{TP}{TP + FP} \]

\textbf{Recall:}
Recall in this context is defined as the number of true positives divided by the total number of elements that actually belong to the positive class (i.e. sum of true positives and false negatives, which are items which were not labelled as belonging to the positive class but should have been.) The Recall can be calculated as:

\[ \text{Recall} = \frac{TP}{TP + FN} \]

\textbf{Accuracy:}
The percentage of the predicted values that match with the expected values for the given data. The best system is that having the high Accuracy, High Precision and High Recall value.

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

IV. RESULT AND DISCUSSION
The implementation of algorithm is done in open source tool known as WEKA 3.2

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 such as:

- High level control variables
- State variables
- Low level control variables
Thereafter, \textit{k-means} clustering algorithm is applied on the dataset. In the \textit{k-means} clustering algorithm the value of $K$ is set to 2 means the total number of clusters that \textit{k-means} clustering algorithm going to generate will be 2.

The Text view of Cluster Assignment is shown in Fig.3. The figure shows that the 15 examples are assigned to cluster 0, 9 examples are assigned to cluster 1.

![Cluster Model](image)

**Cluster 0: 15 items**  
**Cluster 1: 9 items**  
**Total number of items: 24**

When \textit{k-means} algorithm is applied, the various parameters used are shown below:

![Parameter Table](image)

V. CONCLUSION

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. As evidenced by the results, \textit{k-means} Clustering algorithm is proved to be best as compared to the Multi-preceptron algorithm for evaluating the success of software reuse in an organization. It is concluded that for non linear and complex engineering applications involving decision and analysis by and large \textit{k-means} clustering is an efficient technique.

REFERENCES