International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-4, November 2019

MRCC-Measuring Reusability Based on Coupling and Cohesion Methodology

G.Maheswari, K.Chitra

Abstract: ***The component based reusability framework is a procedure of reusing pre-manufactured programming segments to assemble new programming and upgrading the exactness of the techniques for deciding them. The software reuse is principally used to diminish time and spares assets in programming improvement. The product segments are shared and help to lessen the expense of advancement. In existing framework a drawback is the product segment dependent on their nonfunctional reasonableness has been dismissed to an incredible territory. This paper proposes another arrangement of measurements of Fusing and Stability (CC) to determine code scalability frameworks. Coupling and cohesion measures are utilized in different exercises, for example, sway investigation, evaluating the issue inclination of classes, issue forecast, re-modularization, distinguishing of programming part, structure designs, surveying programming quality and so on. It is achievable to infer a productive and hearty reusability forecast model utilizing programming measurements. The product measurement is utilized to improve the nature of programming during the advancement procedure. There are three parts of programming measurements are built up they measure how much substances are coupled one another, it assesses coupling and attachment relationship and lessens the multifaceted nature of classes and techniques. The java is utilized as an assessment instrument for estimating programming measurements and approving the measurements. The experimental results demonstrate the new measurements are reliably prevalent at estimating and positioning the application part of reusability.***

Keywords: Pairing, stability, recycling of code, software component, software metrics, prediction model.

I. INTRODUCTION

Software reusability is a credit that alludes to the normal reuse capability of a software part. A segment can be viewed as a free replaceable piece of the application that gives an unmistakable particular capacity. A component can be a coherent bundle of programming that can be autonomously created and conveyed as a unit, and that offers interfaces by which it very well may be associated unaltered with different segments to form a bigger framework. Software reuse improves profitability as well as positively affects the quality and viability of programming items.

Manuscript published on November 30, 2019.
* Correspondence Author
G.Maheswari*, Assistant Professor, Mangayarkarasi College of Arts and Science for Women, Madurai, **gm291276@gmail.com**
Dr. K.Chitra Ph.D ,Assistant Professor, Government Arts College, Melur.

These two parts of a product segment are firmly related. A very much structured part, In which the utility of its numerous subprograms has been sufficiently communicated, it is expected to go on without limitation and will therefore be easier to apply. Storage allocation is based on several core concepts of the object found scheme: pairing and connection. Linkage is the level of interaction between the various subprocesses. On the off chance that they are much related, at that point changes to one are probably going to affect the conduct of others. Consequently free coupling between its subcomponents is an alluring normal for a segment. Union is the degree to which the capacities performed by a subsystem are connected. On the off chance that a subcomponent is in charge of various inconsequential capacities, at that point the usefulness has been inadequately dispersed to subcomponents. Henceforth high attachment is a normal for a very much planned subcomponent. Numerous measurements. It was suggested that polarity and stability be quantified in order to predict the lack of tendency and practicability of software programs. In any scenario, few tests had been completed using pairing and continuity to measure the commercial viability of parts in view of their significant barriers and the difficulties of evaluating the scalability of quadrants.

Cohesion presumes the degree of interaction between class persons whereas bonding essentially means the interconnectivity of components within a system or software. A highly unified team would be a difficult refactoring prospect, for instance It is impossible to break into different groups. The degree to which the material is concerned should be inferred by high convergence or low correlation by their consequences. High consistency, working within [0, 1] reveals just as good a good framework. Scheduling. However, consistency is a percentage of the hardware level within a sole class-doe the technique to claim that it reveals and displays the partnership within an unit. For linking, it is to claim that it reveals associations between a material or package's components or sections. Union measures how a component distinguishes one capability operated with another. Of most assessments, it is common to determine connection by determining how strategies for a category are capable of evaluating case parameters structures. Linkage means how far the one section cares about each other's internal behaviors (components).

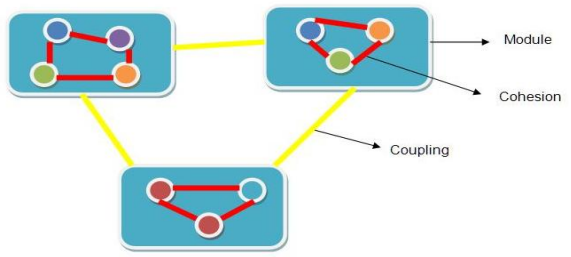

Figure 1: Simple illustration of cohesion and coupling

Characteristics Of Coupling Metrics

Since bonding is the extent for which the structures which create up the conditions are linked, a qualitative calculation would be to provide that one component can be connected to many other individuals. There is an amazing range in all things being equal How does the relation mean, how the counting is performed and also how the amounts are normalized? The features of the measures used in the comprehensive study are listed in Table 1.

Table 1 Characteristic of Coupling Metrics

Name	Explanation
Coupling Between Object Classes (CBO)	The number of groups in which it has been combined is the pairing of a category. A group is assumed to be affiliated with another group in the case that one of its strategies is at most one of its variables or recall. The residual pairing was neglected.
Response for Class(RFC)	A test of a group's response collection; the response set includes all the tactics of the group in relation to the different tactics specifically dreamed up by these same strategies.
Coupling Factor (CF)	Classes are combined when the other class employs methods and instance variables. CF seems to be the sum of total classification groups multiplied by both out of the amount of group sets for a content scheme.
Data Abstraction Coupling(DAC)	A DAC is the amount of water of characteristics (collecting strategies and obvious reasons considerations) that have different categories as their various kinds.

Characteristics Of Cohesion Metrics

Cohesion means the resemblance of a class's strategies. It is a relation of the degree at which an item defines the different environments. Lack of consistency shows that the group can be divided into two categories at minimum. Cohesiveness's innate reasoning is represents that how powerful the components that makes up the conditions. Table 2 condenses the qualities of the attachment measurements utilized in the similar examination. The greater part of the measurements assesses reference and conjuring connection between techniques.

Table 2 Characteristics of Cohesion Metrics

Name	Explanation
Lack of Cohesion in Methods (LCOM)	Number of non-comparative strategy combines in a class of sets. A technique is considered to be like another strategy in the event that they use (by referencing) at least one traits in like manner.
LCOM3	Number of associated parts in chart whose vertices are techniques and whose edges interface comparable strategies.
RLCOM	Proportion of number of non-comparative technique sets to add up to number of strategy matches in the class.
Tight class cohesion(TCC)	Proportion of number of comparable technique sets to add up to number of strategy combines in the class.

The rest of this proposed work is illustrated as follows. In Section II describes related works about software reusability using data mining. The proposed MRCC methodology work and algorithm implementation is described in section III. Section IV presents the experimental results. Section V compromises the conclusion.

II. RELATED WORK

Priyalakshmi et al(2018) Code reuse has turned out to be extremely well known among programming designers as of late since it spares time and assets. One of the huge challenges to programming reuse is the time relating to survey the wellness of the reusable code parts. Over the ongoing years, code web crawlers have made earth shattering progression in building up the semantic appropriateness of programming segments for new use situations. In any case, the issue of assessing programming segments dependent on their nonfunctional appropriateness has been ignored to an enormous degree. The support and reusability of programming frameworks are profoundly affected by the basic properties of framework classes like multifaceted nature, measure, coupling, attachment, and so on. The nature of article arranged code or plan ancient rarities is normally estimated by investigating the structure of these antiquities regarding the interdependencies of classes and segments just as their inner components. In this paper, we play out an exact investigation on Python bundles for the two estimates to be specific coupling and attachment. The coupling score of a module is registered as module imports and the attachment score of a module is assessed as call reliance among classes and worldwide elements of the module. At last, the proposed MRCC work assesses a bundle regarding reusability score which is a total score of the coupling scores and union scores of the considerable number of modules inside the bundle. The investigation has assessed 15 unique bundles and five distinct arrivals of one single bundle for reusability. We have experimentally tried that the Halstead's exertion metric is contrarily relative to the reusability score. The reusability score was approved utilizing four code detuners. The proposed work was contrasted and the current measurements to be specific cyclomatic multifaceted nature and practicality Index demonstrating attractive outcomes. Matheus Paixao et al(2018) Search-based programming designing has been broadly connected to the issue of finding improved measured structures that boost union and limit coupling. Be that as it may, there has, up to this point, been no longitudinal investigation of designers' usage, over a progression of successive discharges. In addition, results approving whether designers regard the wellness capacities are rare, and the conceivably problematic impact of inquiry based remodularization is generally neglected. We present an experimental investigation of 233 successive arrivals of ten distinct frameworks; the biggest observational examination announced in the writing up until this point, and the main longitudinal investigation. Our outcomes give proof that engineers do, in fact, regard the wellness capacities used to enhance attachment/coupling (they are measurably fundamentally superior to self-assertive decisions with $p \ll 0.01$), yet they likewise leave significant space for further improvement (union/coupling can be improved by 25% by and large).

In any case, we likewise report that improving the structure is very problematic (by and large over 57% of the structure must change), while our outcomes uncover that designers will in general maintain a strategic distance from such disturbance. In this manner, we present and assess a multi objective (MO) transformative methodology that limits interruption while expanding union/coupling improvement. This enables engineers to adjust hesitance to disturb existing particular structure, against their contending need to improve union and coupling. The MO approach can discover particular structures that improve the union of designers' usage by 22.52%, while causing an acceptably low degree of interruption (inside that previously endured by engineers).

Hajarisena Razafimahatratra et al(2017) The quality of source code is a significant stake for the development of programming. The source code quality encourages the support and the advancement of the program. In the article situated worldview, a secluded framework is anything but difficult to keep up gratitude to its parts with solid attachment and powerless or low coupling. Powerless coupling is among the key factor to decrease the expense and the due date of the framework creating and keeping up. We propose to assess this factor since the structure of framework. In this paper, the proposed methodology comprises in the programmed identification of coupling type in the item arranged structure level. To do this, we have expounded guidelines to distinguish the most grounded coupling type. It has a place then with the originator to do upgrade for a frail coupling code. Thus, the proposed procedure recognizes the kinds of coupling inside each UML grouping outline object at the structure level. To approve our technique, we led a contextual investigation. At that point, the outcomes got are adjusted in a fluffy design for approval. Lastly, the yield of the fluffy design is a fluffy participation capacity of complete yield coupling. The outcomes demonstrate that our methodology assists creators and programming designers with obtaining a superior nature of programming item.

Amit Ratheea et al(2018) Expanding the product plan quality is a key research challenge in article arranged programming advancement framework. Attachment is one of the key spect that assesses the quality and measured quality of a product framework at the plan level. It makes programming parts that are legitimately reusable to the business as a result of their less reliance on different segments. In this paper, another union measurement for article situated programming, named as Usage Pattern Based Cohesion (UPBC), is proposed which is processed at the module level. This paper considers class as a module at first and in this manner gathering of classes (for example a bundle) is considered as a module with a point of improving by and large attachment. This measurement uses the Frequent Usage Patterns (FUP) separated from various part works associations to catch the cohesiveness of the module. Further, the deliberate attachment worth is utilized to perform bunching of modules so as to expand union and lessening coupling among modules all the while. The grouping is performed by utilizing a recently proposed bunching calculation called FUPClust (Frequent Usage Pattern based Clustering) in light of FUP communications among modules. The proposed methodology is connected to two Java programming frameworks and the outcomes acquired demonstrate a huge improvement in the cohesiveness of the product framework.

Sushma Yadav et al (2014) Programming measurements are fundamental to improve the nature of programming during the advancement procedure. Coupling and attachment measures are utilized in different exercises, for example, sway investigation, evaluating the shortcoming inclination of classes, deficiency forecast, re-modularization, recognizing of programming part, plan designs, surveying programming quality and so on. Low coupling and high attachment are better for good programming quality. Coupling and union measurements can be connected at the early period of the product advancement process. This paper surveys different coupling and attachment measurements for item situated programming.

Mr. Kailash Patidar et al (2013) Many coupling and union measures have been acquainted in different studies with distinguish and measure the structure unpredictability of article arranged frameworks. Enormous quantities of measurements have been fabricated and proposed for estimating properties of item situated programming, for example, measure, legacy, attachment and coupling. The coupling is a significant perspective in the assessment of reusability and practicality of parts or administrations. The coupling measurements discover intricacy among legacy and interface programming. In this paper presents estimations of article after that discover the coupling and union between items, measure the relationship between quantities of classes, check the immediate conditions, backhanded conditions, IO conditions, number of out and in measurements in item situated programming. Estimation is performing among legacy and interface programs. This paper additionally proposes a model to gauge the level of coupling and attachment because of these conditions.

Saurabh Tiwari et al(2018) Coupling and Cohesion are two basic ideas that can be connected to configuration better secluded item situated programming. This examination goes for investigating existing exploration on coupling and union measurements so as to recognize the potential ones and requirements for the future research. A methodical mapping study is introduced to distinguish the well known coupling and union measurements, and their materialness practically speaking. The examination uncovered that the essentialness of coupling and union measurements in different programming improvement exercises has been supported by different scientists. Be that as it may, a few issues, for example, the absence of accessibility of data about the logical uses of these measurements and their various understandings by various analysts should be set out to build up the viable utilization of these measurements.

III. THEORETICAL BACKGROUNDS

Proposed Mrcc Methodology

The flow of proposed MRCC approach is explained in the below figure discuss about the software metrics and measuring of coupling and cohesion software reusability and find the solution for complexity parameters in software metrics.

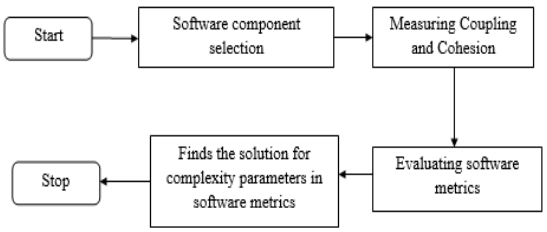

Figure 2 Proposed MRCC Methodology of Coupling and Cohesion

Proposed Coupling Metrics

In order to build a pairing measure that evaluates the degree of correlation, functional unpredictable nature, and conjunctive (i.e. aberrant) fusing between groups, we start with as an organized diagram with consideration to any articles structured computing structure. The groups that form the frame are the graph's vertices. Suppose an implementation of this kind contains a lot of classes C={C 1,C 2..... C m}. Configuration techniques and case considerations in class C I used by class C j for ji (Structure of tactics and category variables in class C I is defined as inaccurate). A boundary from C j to C i persists at that stage if and especially if there is no nullity of every MV (j,i). In this sense, an angle of the map represents the absolute linking of one group to another through class C I inv by the organization of techniques and case variables. The illustration is organized as −MV (j,i) is never inherently equivalent to −MV (i,j) the scheme of all items considering and instance factors may be described in various groups conjured by C j.:

$$MV_j = \cup_{1 \le i \le m} MV_{j,i} \qquad (1)$$

The quantity of techniques and variables in the array a ubiquitous MV (j, I (ubiquitous I j)) represents the degree of explicit coupling from C I to C j. T should be smaller if the group calls for a larger number of other approaches. Nevertheless, this percentage should also reflect the way a group summing up various tactics has a far more significant possibility of inventing approaches from a particular class. In all of these lines, CoupD(i, j) is described as the amount of direct correlation between class C I and C j.

$$CoupD(i,j) = \frac{|mv_{i,j}|}{|MV_i| + |V_i| + |M_i|} \qquad (2)$$

The numerator was an all-out amount of methods used by Class C I i.e. the all-out utility of Class C I i.e. the complete effectiveness of Class C i. It means the CoupD(i, j) is independently for Class Assessment 1 CoupD(i, j) is going to Be constantly shy of what it should do on the grounds that even the exceptional quality would only be achieved in the incredibly unusual case of a group without any persons associated with a single unique category. Make the assumption, therefore, that CoupD(i, j) and CoupD(j, k) has minimal attributes, but that CoupD(i, k) is null. Therefore, regardless of the fact that there will be no instant convergence amongst C I and C k groups, there is still a dependency on C I summoning tactics in C j. Captures strategies in C (k.) Because the value of this dependence is dependent on the dual immediate linkages from which it is generated, a fair estimate is given, CoupD(i, j) = CoupD(j, k). There is still a correlation among several groups if there is indeed a path between one and a next made up margins where CoupD properties are all non-zero. The consistency of the correlation is the product of either of those CoupD attributes. This is how we define CoupT(i, j, p), the prepositional pairing between C I and C j classes due to a massive particular manner p,as

$$CoupT(i,j,p) = \prod_{e_{s,t} \in p} CoupD(s,t) \qquad (3)$$

$$= \prod_{e_{s,t} \in p} \frac{|MV_{s,t}|}{|MV_s| + |V_s| + |M_s|} \qquad (4)$$

E (s, t) also shows the edge among vertices s and t. CoupT for the subsequent CoupD coupling, that corresponds to the first and subsequent lengths because, provided that the CoupD figures are significantly below one, predicate linkages would usually have smaller estimates due to excessive further ways.

As a principle, there could be more then once option to have an anti-zero CoupT confidence between every other groups. To solve a few current issues, a simpler straight-to-earth solution is used to split the route with the greatest CoupT price and therefore define Coup(i, j) the consistency of the connection between both the two groups, C I and C j:

$$Coup(i,j) = CoupT\left(i,j,p_{max}(i,j)\right) \qquad (5)$$

Where p max I j)=arg arg Ã rmmax (pi) CoupT(i, j, p) and Ã rm are all forms from C I to C j setup. The last move was to use this as a rationale for a fraction of the material new framework full linking. This is done immediately by summarizing each of the linkages of all group events and separating them by the actual number of such combinations.

Proposed Mrcc Cohesion Metrics

Measures As mentioned earlier, our suggested MRCC category connection metrics rely on the notion of friendship to the methodology. It is said that techniques are equal when they mask the configurations for periodic variables they control. They also assign their linking test to be the vertices of either the approaches for a group of a diagram. Assume that a group has a number of techniciansM={M 1,M (2....) M m} and let V j{V (j,1,) V (j,2,)..... V (j,n)} be the examples for methodM j. The edge of M j to M I exists at a certain point only if V (j) V I is not invalid. In all of these rows, an angle of both the outline represents the relative proximity of the tactics we exchange at any price. In action, one case parameter. Unlike the coupling map, in view of the fact that convergence is a symmetric relation, the similarity diagram is undirected. The next move is to add a number of each edge which represents the extent to which the dual approaches shares incidence influences that have been basically SimD(i, j). The possible MRCC ratio of two strategies ' direct similarity, M I to M j, is described as

$$SimD(i,j) = \frac{|V_i \cap V_j|}{|V_j \cap V_i|} \qquad (6)$$

Where $i \neq j$ ($SimD(i,j)$ is characterized to zero). The denominator is the absolute number of factors utilized in two techniques and the numerator is the regular factors shared by two strategies. The standard deviation to be the all-out efficiency of M I and M j method. SimD(i, j) is thus free of both the size of either the technique. The rise in the formula to add backhanded correlation is close to what we used for round about pairing.

The value of proximity that a path offersThe product of SimD

calculations of the boundaries that made up the same way among two approaches. Throughout this way, SimT(i, j) is defined as the predicate similarity amongst M I and M j approaches due to a massive particular way p

$$SimT(i,j,p) = \prod_{e_{s,t} \in p} \quad SimD(s,t) = \prod_{e_{s,t} \in p} \frac{|V_s \cap V_t|}{|V_s \cup V_t|}$$

The outcome of SimD measurements of both the edges that create up with the way through several approaches. In this form, SimT(i, j) is defined as the predicate relation around M I and M j approaches due to a massive specific manner p., $Sim(i,j)$

$$Sim(i,j) = SimT\left(i,j,p_{max}(i,j)\right) \quad (8)$$

Wher $p_{max}(i,j) = arg\ max_{p \in \prod}\ SimT(i,j,p)$ and $\prod$ is the arrangement of all ways from M_i to M_j.

This metric is used to provide a measure of the group connection, ClassCohT, by summarizing the correlations of all tactical games and dividing them by the amount of such combinations:

$$ClassCohT = \frac{\sum_{i,j=1}^{m} \quad Sim(i,j)}{m^2 - m} \quad (9)$$

This metric is used to provide a measure of the group connection, ClassCohT, by summarizing the correlations of all tactical games and dividing them by the amountof such combinations:

$$ClassCohD = \frac{\sum_{i,j=1}^{m} \quad SimD(i,j)}{m^2 - m} \quad (10)$$

Where, m is the number of methods in the class.

Figure 3 Is a template of the proposal MRCC adjustments that (in view of the variable technique) find the extension or modification of strategic technological collaborations. Nevertheless, the concept of combining varying-strategy measures the organization's nonattendance although the last measures its nature. The variables are the I, j, k and l, while the techniques are M1, M2 and M3

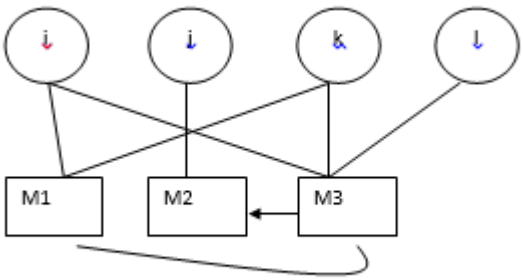

Figure 3 Illustration of class cohesion measure indicating variable-method interaction and method-method interaction.

IV. EXPERIMENTAL RESULTS

In the exploratory outcomes, the developer was required to make basic expansions to Java parts recovered utilizing our segment internet searcher. In perspective on the current experimental help for the view that coupling and attachment measurements are decidedly related with deficiency inclination. Basically all that is required is to take a lot of parts and approach able software engineers to adjust them for use in bigger frameworks. This trial study is in this way worried about assessing the measurements as indicators of the adjustment exertion required. To assess the exhibition of the different measures in foreseeing reusability utilized spearman rank connection. There was significant variety in part size inside each gathering (see Table 3). Altogether, the 65 segments contained 1974 classes.

Table 3 Number of Code and Size of Component Group

Component Group	Lines of Code	File Size
HTML Parser	1094	78k
Lexical Tokeniser	1348	87k
Barcode Generator	2213	161k

Here registered the Spearman rank connection coefficients between the rankings dictated by NLOC and those created by the different coupling and attachment measures.

Table 4: Results of Coupling Metrics Measures

Measures	HTML Parser	Lexical Tokeniser	Barcode Generator
RFC	0.789	0.732	0.678
DAC	0.509	0.717	0.700
CF	0.787	0.298	0.768
CBO	0.465	0.123	0.478

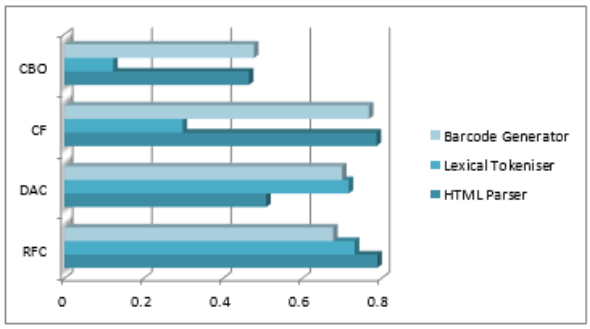

Figure 4 Performance Analyses of Coupling Metrics

Table 4 and figure 4 demonstrates the outcomes got for the coupling measurements. The request where they show up in the table is dictated by their normal execution over each of the three part gatherings. The lion's share were measurably noteworthy, a significant number of them at the 1% level. In any case there are impressive contrasts in the level of connection accomplished. The RFC is contrasted as best methodology contrasting and different parts in coupling.

Table 5: Results of Cohesion Metrics Measures

Measures	HTML Parser	Lexical Tokenize	Barcode Generator
TCC	0.522	0.767	0.835
RLCOM	0.342	0.787	0.700
LCOM3	0.808	0.830	0.100
LCOM	0.218	0.005	0.509

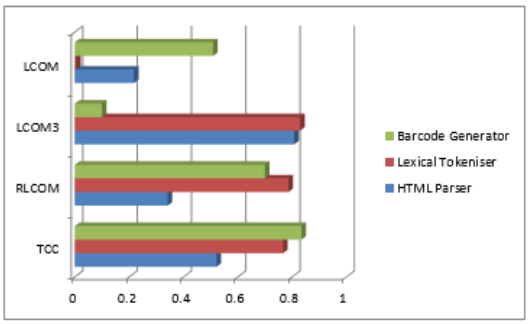

Figure 5: Performance Analyses of Cohesion Metrics

Table 5 and Figure 5 show the results of the calculations of the connection. It should be seen that somehow the percentages of LCOM, LCOM3, and RLCOM are lack of closeness. These are unlike contact comparable and therefore The effort needed to change the part should be in direct proportion.

RLCOM and TCC conduct higher performance to LCOM3 and LCOM, which shows that the size of the strategies may influence estimate analysis. Apparently, such results indicate that suggested Pairing as well as reasonableness calculations by MRCC are amazing predictors of the effort needed to make simple changes to Java parts collected from the Web.

V. CONCLUSION

Reusability has an extraordinary significance in programming building and its utilization in programming designing is expanding step by step. However, the issue is that there is no appropriate method to remove reusable segments or articles. To illuminate this issues Spearman rank relationship is accustomed to deciding the reusable segments from existing measurements like union and coupling. In this closing segment we initially think about the commitments of the proposed MRCC measurements. At that point we survey whether they are satisfactory for the reason for which they were created: delivering a reusability positioning for parts recovered from the Internet. This application is created in java. We get 65 programming segments from various site and get our outcomes as charts.

Our future meant to sum up proposed bunching process. In proposed system programming segments are bunched with explicit configuration so later on we need to group programming segments without explicit organization and parameters for grouping programming segments extricate at runtime.

REFERENCES

1. Mr. Kailash Patidar, Prof.RavindraKumar Gupta, Prof.Gajendra Singh Chandel "Coupling and Cohesion Measures in Object Oriented Programming" International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 3,2013,pp-517-521.
2. G. Priyalakshmi and R. Latha "Evaluation of Software Reusability Based on Coupling and Cohesion" International Journal of Software Engineering and Knowledge Engineering Vol. 28, No. 10,2018, pp. 1455-1485
3. Joshi P, Joshi RK. Concept analysis for class cohesion. Proceedings of the 13th European Conference on Software Maintenance and Reengineering. Kaiserslautern, Germany. 2009. p. 237–40.
4. Matheus Paixao, Mark Harman,Yuanyuan Zhang, Yijun Yu "An Empirical Study of Cohesion and Coupling: Balancing Optimization and Disruption" IEEE Transactions on Evolutionary Computation Volume: 22, Issue: 3,2018,pp- 394 – 414.
5. Hajarisena Razafimahatratra ; Thomas Mahatody ; Josvah Paul Razafimandimby ; Sabin Mihai Simionescu "Automatic detection of coupling type in the UML sequence diagram" 21st International Conference on System Theory, Control and Computing (ICSTCC) 2017.
6. AmitRatheeaJitender KumarChhabra "Improving Cohesion of a Software System by Performing Usage Pattern Based Clustering" Procedia Computer Science Volume 125, 2018, Pp-740-746.
7. Saurabh Tiwari,Santosh Singh Rathore "Coupling and Cohesion Metrics for Object-Oriented Software: A Systematic Mapping Study "11th Innovations in Software Engineering Conference Issue 8,2018.
8. Gui Gui, Paul. D. Scott "Measuring Software Component Reusability by Coupling and Cohesion Metrics" Journal of Computers, Vol. 4, Issue. 9, 2009,pp-797-805.
9. Adekola, O.D, Idowu, S.A , Okolie, S.O, Joshua, J.V, Akinsanya, A.O, Eze, M.O, EbiesuwaSeun "Software Maintainability and Reusability using Cohesion Metrics" International Journal of Computer Trends and Technology (IJCTT) – Volume 54 Issue 2-2017-pp-63-73.
10. Pooja Rana, Rajender Singh "Comparative Analysis of Cohesion Metrics For Component Based Software System" Journal of Theoretical and Applied Information Technology Vol.96. Issue 14, 2018.
11. Karminder Kaur and Gurjot Kaur "Component Reusability of a Software System based on Cohesion and Coupling" Indian Journal of Science and Technology, Vol 9,Issue 27,2016,pp-1-6.
12. Neelamadhab Padhy, R.P. Singh, Suresh Satapathy "Software reusability metrics estimation: Algorithms, models and optimization techniques" Computers & Electrical Engineering Vol.69, 2017.
13. Matheus Paixao, Mark Harman, Yuanyuan Zhang, and Yijun Yu "An Empirical Study of Cohesion and Coupling: Balancing Optimisation and Disruption" IEEE Transactions In Evolutionary Computation 2017
14. Hajarisena Razafimahatratra, Thomas Mahatody, Josvah Paul Razafimandimby "Automatic Detection of Coupling Type in the UML Sequence Diagram" 21st International Conference on System Theory, Control and Computing (ICSTCC),2017,pp-635-640.

AUTHORS PROFILE

G.Maheswari , Assistant Professor, Mangayarkarasi College of Arts and Science for Women, Madurai, **gm291276@gmail.com**

Dr. K.Chitra Ph.D ,Assistant Professor, Government Arts College, Melur.