Available online at http://acfa.apeejay.edu
APEEJAY JOURNAL OF COMPUTER SCIENCE AND APPLICATIONS
ISSN:0974-5742(P)
Vol. 3, January 2015

Estimating Software Reusability from OO Metrics using Fuzzy Logic

Pooja Dhand[1], Parwinder Kaur Dhillon[2], Jagmohan Mago[3]
[1]Assistant Professor, Apeejay College of Fine Arts, Jalandhar, [2]Assistant Professor, Apeejay College of Fine Arts, Jalandhar, [3] Assistant Professor, Apeejay College of Fine Arts, Jalandhar
[1]pooja_dhand@hotmail.com, [2]parwindhillon@gmail.com, [3]magojagmohan@gmail.com

Abstract
With the rise of the OO software development the traditional software metrics have proved inadequate to measure such systems. This has led to the development of metrics suited to the OO model. Chidamber and Kemerer proposed a set of six metrics to measure the OO aspects in order to help the users in understanding the complexity of object oriented design and in predicting the external software quality attributes such as software defects, testing, and maintenance effort. In this paper, we have used fuzzy logic to evaluate the quality of an OO system in terms of reusability. We have tried to analyze and interpret the impact of six CK metrics on the reusability of the system. Simulation results demonstrate how the reuse factor varies with respect to the metrics viz. WMC, DIT, NOC, CBO, LCOM and RFC.
Keywords: *Software Reuse, CK Metrics, OO Design, Fuzzy Logic*

1. Introduction

Over the years there is an increase in object-oriented approach to software engineering. This is because it improves the development of software, including such factors as high reusability and increased extensibility [1,9]. Moreover the developer is able to create systems which are robust, adaptable and easily maintainable. To help the designers and developers achieve these goals, researchers have proposed a large number of metrics e.g. CK metrics [3], MOOD metrics [2] etc. Amongst all the widely used and accepted set of metrics is the set proposed by Chidamber and Kemerer (CK) [3]. With such vast amount of metrics available it becomes important to understand them in order to acquire a precise and precise understanding of the software being evaluated.

An interdependence of modules in design makes it rigid, fragile and difficult to reuse. A single change in an interdependent system leads to a cascade of changes in its dependent modules. This increases the cost of the software. Fragility decreases the reliability of the design and maintenance team. Users and managers are not able to estimate the quality of their product. A design is difficult to reuse when the necessary sections of the design are greatly dependent on other modules which are not wanted [4]. Designers tasked with investigating the design to see if it can be reused in a different application may be impressed with how well the design would do in the new application. Nonetheless, given the different metric suites to measure the reusability of design, the developers are still confronted with the problem of interpreting these metric values [12, 17].

A mechanism is needed that predicts quality of a software system by evaluating its reusability the use of metrics. In this paper, we have taken in an effort to apply object oriented metrics as a predictor for software reusability of the underlying system. An integrated model using fuzzy logic in MATLAB is proposed which takes as input the metrics proposed by Chidamber and Kemerer. Fuzzy logic provides a basis to define a language, to associate a meaning with each aspect of the language and to compute these expressions [1, 5]. Software developers can express their intuition of an artifact type using their natural language and this intuition can be modeled and maintained on the overall development process. Besides, the software engineers' perception is not suppressed by the necessity to fit restrictive input values imposed by methodological rules. Capturing as much as possible method developers' intuition and consequently the software engineers' perception, reduce the loss of information and improves the quality of the overall development process. Also, the influence of contextual factors on the validity of methodological rules can be controlled by adapting the meaning associated with each linguistic expression [21].

The paper is organized as follows: Section 1 gives introduction to the problem, Section 2 reviews the available literature for using OO metrics as quality predictors, Section 3 presents the description of the empirical study viz. the data sets used, OO metrics used etc.; Section 4 presents the system model in detail; Section 5 discusses the performance results and finally Section 6 concludes with the future enhancements that can be made to the proposed model.

2. Related Work

Sahraoui et al. in [10] illustrated a novel approach to build rule-based predictive model based on fuzzy logic. The approach seeks to connect the cognitive gap that may exist between the antecedent and the consequent of a rule by turning the latter into a range of sub rules that account for domain knowledge. The whole framework is evaluated on a set of OO applications. P. Sandhu et al. In [15] an algorithm has been suggested to predict reusability. In this model the inputs were given to Neuro-fuzzy system in form of tuned WMC, DIT, NOC, CBO, and LCOM values of the OO software component and output can be obtained in terms of reusability.

S. Anju et al in [22] tuned CK metric suit i.e. WMC, DIT, NOC, CBO and LCOM, to obtain the structural analysis of OO-based software components. An algorithm has been proposed in which the inputs can be given to K-Means Clustering system in form of tuned values of the OO software component and decision tree forms for the 10-fold cross validation of data to evaluate the in terms of linguistic reusability value of the component. The developed reusability model has produced high precision results as desired. J. Mago in [24] proposed a model based on fuzzy logic as an integrated means to provide an interpretation of the OOD metrics of the CK metric suite.

3. Description Of Empirical Study

3.1 Data Set

We have used the data sets available from the PROMISE data repository. Data sets from PROMISE data repository were used to test the efficacy of the proposed model. PROMISE data repository is a collection of publicly available datasets and tools to serve researchers in building predictive software models (PSMs) and software engineering community at large. The repository is created to encourage repeatable, verifiable, refutable, and/or improvable predictive models of software engineering. We have used the 'Class-level data for KC1' public data set to pick the values of CK metrics

3.2 Object Oriented Metrics Considered

In order to assist managers and developers achieve high quality OO software, a variety of software metrics have been proposed in the literature. Chidamber and Kemerer, in 1994 [3], developed a set of six metrics to identify certain design and code traits in OO software.

a) Weighted Methods per Class (WMC)
b) Depth of Inheritance Tree (DIT)
c) Number of Children (NOC)
d) Coupling between Object Classes (CBO)
e) Response for a Class (RFC)
f) Lack of Cohesion of Methods (LCOM)

4. Proposed Model

In the proposed model, we develop two different fuzzy logic controllers to evaluate the value of Reuse Factor to determine the quality of the software under consideration. In order to reduce the number of rules and system complexity, these two fuzzy logic controllers are combined in a parallel fashion. The outputs of these two FLCs are then used to compute the weighted reuse factor. The Mamdani type FIS is incorporated with carefully designed rules. Figure. 1 shows the overall design of these FLCs that are used to determine the reusability. In the following subsections, the details of all the fuzzy logic controllers are provided.

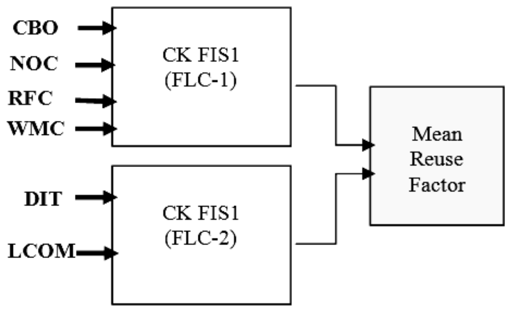


Figure. 1. Fuzzy Logic Controllers for Reuse Factor

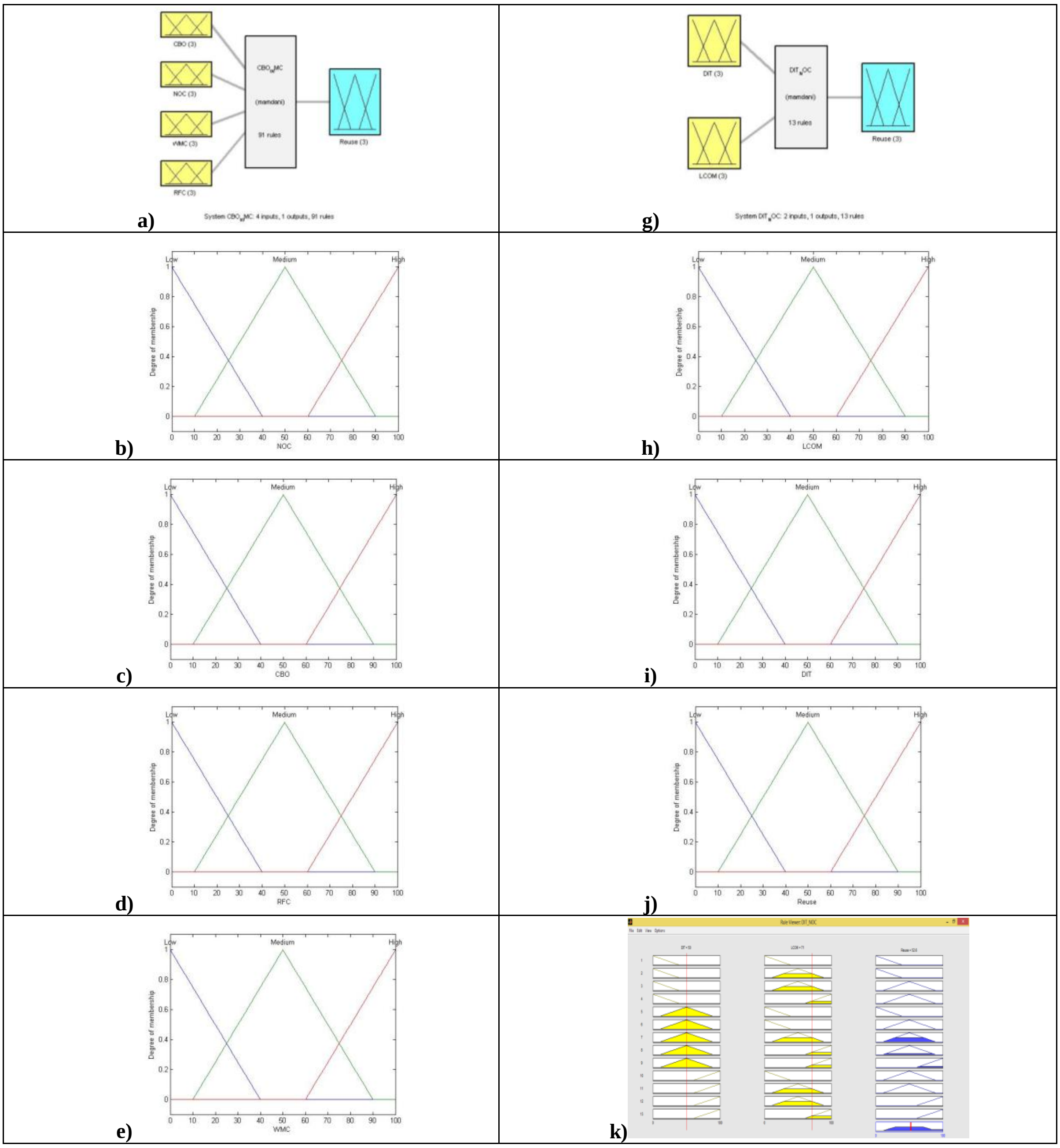
We have extended the existing linguistic modeling technique by making it more flexible with the use of double consequent weighted rules. A double consequent weighted rule is the one where for same set of antecedents we may have two consequents with an importance degree associated with each fuzzy rule. The weighted double-consequent fuzzy rule [28] has the following structure:

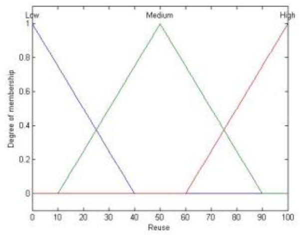
$$\text{If } x_1 \text{ is } A_1 \text{ and } \ldots.\ X_n \text{ is } A_n \text{ then Y is } \{C_1 \text{ with } [w_1],\ C_2 \text{ with } [w_2]\}$$

Where w_1 and w_2 are the weights associated with the consequents C_1 and C_2.

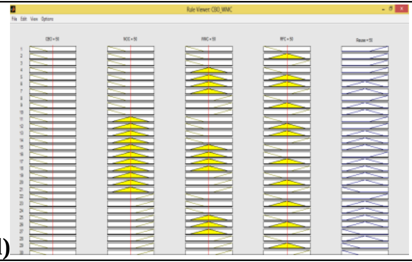
4.1 Design of Fuzzy Logic Controllers

FIS controller1 uses the four of the CK metric (WMC, CBO, RFC and NOC) to predict the reusability of Object Oriented Design. FLC 2 predicts the reuse factor from DIT and LCOM. The fuzzy inference model used is MAMDANI. The input values of the metrics for a class are fed into the fuzzy systems. Depending upon the input values of the metrics, some rules out of the total 91 rules from the knowledge base of FLC1 gets fired and 13 rules from the knowledge base of FLC2 gets fired. The Defuzzification technique used is '*Centroid of area*' method. We have used triangular membership functions. The input variable has three membership functions (*Low*, *Medium*, and *High*) and the output variable also has three singleton membership functions (*Low*, *Medium*, and *High*). The output variable reflects the degree of reuse of the class. The system model, input membership for WMC, DIT, RFC, NOC, CBO and LCOM for the FLC1 and FLC2 are shown in Figure. 2. (a) - (l).





f)



l)

Figure. 2. (a)-(e) FLC1, Input membership Functions of FLC1; **(f)** Output membership function of FLC1; **(l)** Rule Viewer of FLC1. **(g) -(i)** FLC2, Input membership Functions of FLC2; **(j)** Output membership function of FLC2; **(k)** Rule Viewer of FLC2.

4.2 Design of Knowledge Base for FLC1 and FLC2

The knowledge base of FLC1 consists of a total of 91 rules. The inference method used is Mamdani. Few of the Rules are listed in table 1 as shown below.

Table 1 Rules of FLC1

Rule No.	CBO	NOC	WMC	RFC	Output
1	Low	Low	Low	Low	High
4	Low	Low	Medium	Low	High
18	Low	Medium	Medium	High	Medium
21	Low	Medium	High	High	Low
36	Medium	Low	Medium	Low	High
68	High	Low	Medium	High	Low
91	High	High	High	High	Low

The knowledge base of FLC2 consists of a total of 13 rules. The inference method used is Mamdani. The complete set of rules is listed in Table 2. The rules 2,3,5,6,8,9,11,12 are the double consequent weighted fuzzy rules with weight being 0.5.

Table 2 Rules of FLC1

Rule No.	DIT	LCOM	Output
1	Low	Low	Low
2	Low	Medium	Low
3	Low	Medium	Medium
4	Low	High	Medium
5	Medium	Low	Low
6	Medium	Low	Medium
7	Medium	Medium	Medium
8	Medium	High	Medium
9	Medium	High	High
10	High	Low	Medium
11	High	Medium	Medium
12	High	Medium	High
13	High	High	High

5. Performance Analysis

The proposed reuse prediction model is implemented in MATLAB (R2011a). The fuzzy logic toolbox is used to implement the different FLCs that are used in this research work. Data sets from the PROMISE data repository were used to examine the efficacy of the proposed model. We have used the ‘Class-level data for KC1’ public data set.

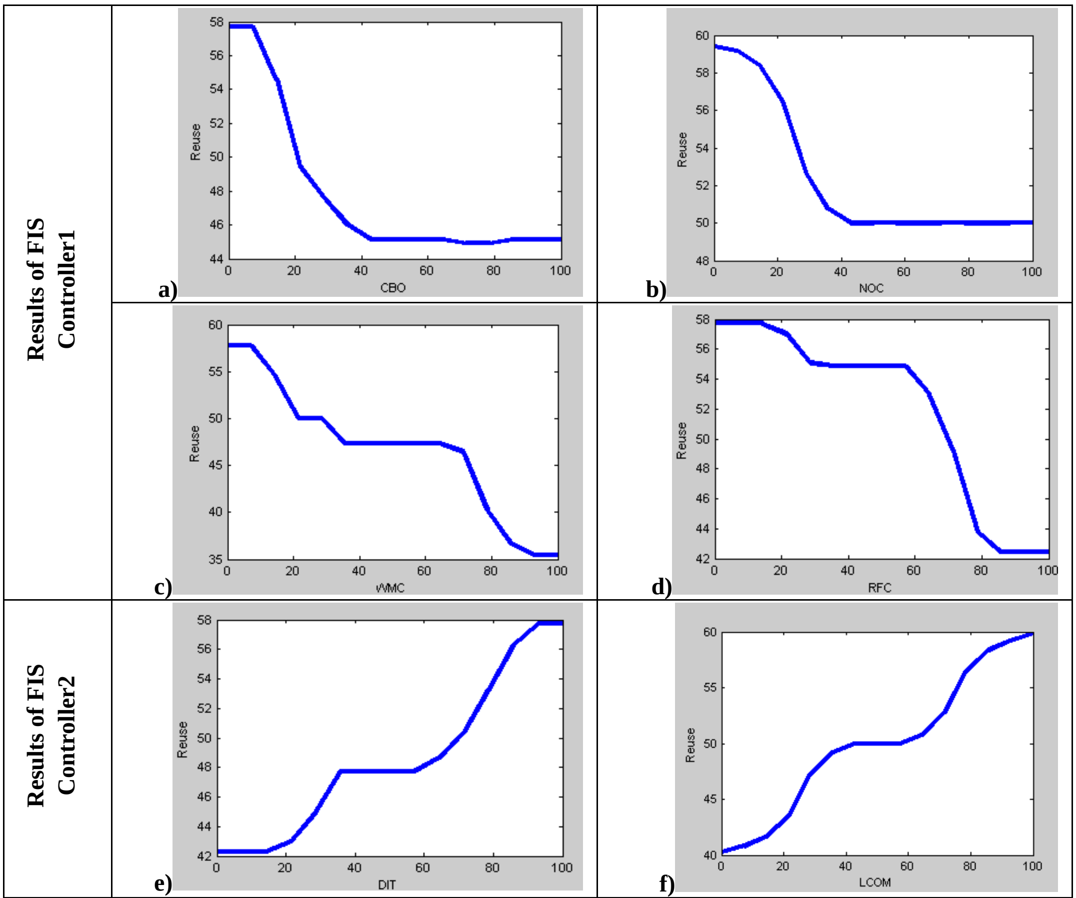


Figure. 3. (a) Graph of Reuse with CBO **(b)** Graph of Reuse with NOC **(c)** Graph of Reuse with WMC **(d)** Graph of Reuse with RFC **(e)** Graph of Reuse with DIT **(f)** Graph of Reuse with LCOM

Extreme coupling between object classes is detrimental to reuse. The graph in Figure 3 (a) shows the relationship between the input variable ‘CBO’ and output variable ‘Reuse’. It shows that as the value of CBO increases the reuse factor of the class decreases. A class with a large number of children is more complex and fault-prone, leading to difficulty in reuse. The graph in Figure 3 (b) shows the relationship between input variable ‘NOC’ and output variable ‘Reuse’. It shows that as the value of NOC increases the reuse factor of the class decreases. The graph in Figure 3 (c) shows the relationship between the input variable ‘WMC’ and output variable ‘Reuse’. High values of WMC signify that the class has a bigger impact on its children and becomes more application specific thereby reducing the possibility of reuse. The large RFC value indicates limited reuse. The graph in Figure 3 (d) depicts the relationship between input variable ‘RFC’ and output variable ‘Reuse’. The deeper a specific class is within the hierarchy, the bigger the potential reuse of inheritable methods. The graph in Figure 3(e) shows the connection between input variable ‘DIT’ and output variable ‘Reuse’. It shows that as the value of 'DIT' will increase the reuse factor of the class also will increase. The graph in Figure 3(f) shows the relationship between input variable ‘LCOM’ and output variable ‘Reuse’. It shows that as the value of LCOM increases the reuse factor of the class also increases. High cohesion implies simplicity and increases reusability.

In the Table 3 below, we present the empirical validation of the proposed model for five classes whose data set is collected from PROMISE data repository as mentioned in section 3.1. This data set predicts whether a class is defective or not. Class1, class2 and class3 are defective ones whereas class4 and class5 are non-defective. The results of table 1, clearly show that the reuse factor of class4 is the highest due to a high level of cohesion among its methods. Also, it has a lower amount of coupling. Similarly for class1, it has a high WMC, CBO and RFC values which increases its complexity and is detrimental to make it effective for reuse.

Table 3 Empirical analysis of the proposed model

Metrics	Class 1	Class2	Class3	Class4	Class5
WMC	73	30	99	8	6
NOC	0	0	0	2	0
DIT	4	4	1	1	1

RFC	**100**	**78**	**99**	**8**	**6**
CBO	**24**	**19**	**21**	**4**	**4**
LCOM	**100**	**100**	**1**	**75**	**0**
O/P FLC1	**42**	**50.6**	**44.4**	**86.5**	**86.7**
O/P FLC2	**50**	**50**	**13**	**45.7**	**13**
Weighted Reuse	**46%**	**50.3%**	**57.4%**	**66.1%**	**49.85%**

CONCLUSION AND FUTURE SCOPE

The proposed model developed using fuzzy logic uses the metrics given by Chidamber and Kemerer. The model can be effectively used for predicting the degree of reusability of a class in the early phases of SDLC which will result in minimizing the time and effort of the software developers. The study can be extended to deal with object oriented design specifications. More combinations of the different available metrics can be integrated depending upon the requirements of the user. We used the six metrics of CK suite, correlation of other metrics, can also be examined and they can also be used to estimate the reusability. We have used fuzzy logic approach other approaches like neural networks, case based systems can also be used to make the system more effective. We can also find the solution to other inconsistencies to which the solution has not been proposed yet.

References

[1] Zadeh, L.A. (1965). Fuzzy sets, information and control. 8: 338-353.
[2] Abreu, F.B. and Carapua, R. (1994). Candidate metric for OOS within taxonomy framework. Journal System. Software. 26: 1.
[3] Chidamber, S.R. and Kemerer, C.F. (1994). A metrics suite for object oriented design. IEEE Trans. On software engineering. 20(6): 476-493.
[4] Martin, R. (1994). OO design quality metrics. An analysis of dependencies.
[5] Basili, V.R., Briand, L.C. and Melo, W.L. (1996). A validation of object oriented design metrics as quality indicators. IEEE Trans. on software engineering. 22(10): 751-761.
[6] Rosenberg, L.H. and Hyatt, L. (1997). Software quality metrics for object oriented environments. Crosstalk J.
[7] Benlarbi, S., Emam, K.E. and Geol, N. (1999). Issues in validating object-oriented metrics for early risk prediction. Proc. of 10th Int. Symp. On software reliability engineering (ISSRE'99). Boca.
[8] Kamiya, T., Kusumoto, S. and Inoue, K. (1999). Prediction of fault-proneness at early phase in object-oriented development. Proc. of 2nd IEEE Int. Symp. On object oriented real-time distributed computing (ISORC '99):253-258.
[9] Pressman, R.S. (2001). Software engineering-A practitioner's approach. McGraw-Hill international edition. 5th edition.
[10] Li, Bixin. (2001). A hierarchical slice-based framework for object-oriented coupling measurement. Turku Centre for Computer Science.
[11] Sahraoui, Houari A., et al. (2002). A fuzzy logic framework to improve the performance and interpretation of rule-based quality prediction models for oo software. Computer Software and Applications Conference. Proceedings. 26th Annual International. IEEE.
[12] So, Sun Sup, Sung Deok Cha, and Yong Rae Kwon. (2002). Empirical evaluation of a fuzzy logic-based software quality prediction model. Fuzzy Sets and Systems127.2: 199-208.
[13] Menzies, T., Ammar, K., Nikora, A. and Stefano, S. (2003). How simple is software defect prediction? J. Empirical Software Engg.
[14] Subramanyam, R. and Krishnan, M.S. (2003). Empirical analysis of CK metrics for object-oriented design complexity: Implications for software defects. IEEE Trans. Software Engg. 29(4): 297-310.
[15] Aggarwal, K.K., Singh, Y., Kaur, A. and Malhotra, R. (2005). Software reuse metrics for object-oriented system. Proc. of 3rd ACIS Int. Conf. on software engineering research, management and applications (SERA'05). IEEE Computer Soc. pp.48-55.
[16] Sandhu, Parvinder Singh, and Hardeep Singh. (2006). Automatic reusability appraisal of software components using neuro-fuzzy approach." International Journal Of Information Technology 3.3: 209-214.
[17] Seth, Kirti, Arun Sharma, and Ashish Seth. (2009). Component Selection Efforts Estimation–a Fuzzy Logic Based Approach. International Journal of Computer Science and Security,(IJCSS) 3.3 : 210-215.
[18] Khalsa, S.K. (2009). A Fuzzified approach for the prediction of fault proneness and defect density. Proc. of the World Congress on Engineering. Vol. I. WCE 2009. July 1-3, London, U.K.
[19] Dubey, S.K. and Rana, A. (2010). A comprehensive assessment of object-oriented software systems using metrics approach. IJCSE. 2: 2726-2730.
[20] Chandra, E. and Linda, P.E. (2010). Assessment of software quality through object oriented metrics. CIIT Int. J. Software Engg. 2: 2.

[21] Bhatnagar, R., Bhattacharje, V. and Ghose, M.K. (2010). A proposed novel framework for early effort estimation using fuzzy logic techniques. Global J. Comp. Sci. Technol. 10: 14.
[22] Malhotra, R., Kaur, A. and Singh, Y. (2010). Empirical validation of object-oriented metrics for predicting fault proneness at different severity levels using support vector machines. Int. J. Syst. Assurance Engg. Management. 1(3): 269-281.
[23] Shri, Anju, et al. (2010). Prediction of Reusability of Object Oriented Software Systems Using Clustering Approach. World Academy of Science, Engineering and Technology : 853-858.
[24] Mago, Jagmohan, and Parwinder Kaur.(2012). Analysis of Quality of the Design of the Object Oriented Software using Fuzzy Logic. International Conference on Recent Advances and Future Trends in Information Technology.
[25] Saxena, P. and Saini, M. (2011). Empirical studies to predict fault proneness: A review. Int. J. Computer Appl. 22(8): 41-45.
[26] Malhotra, R. (2012). A defect prediction model for open source software. Proc. of the World Congress on Engineering. Vol. II. July 4-6. London (UK).
[27] Handa, A. and Wayal, G. (2012). Software quality enhancement using Fuzzy logic with object oriented metrics in design. Int. J. Comp. Engg. Technol. (IJCET). 3(1): 169-179.
[28] Alcalá, Rafael, et al. "Linguistic modeling with weighted double-consequent fuzzy rules based on cooperative coevolutionary learning." Integrated Computer-Aided Engineering 10 No. 4: pp. 343-355. 2003