Volume 4, Issue 3, March 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Ranking and Clustering of Software Cost Estimation Models

Ms. Vijaya Wable*, Assist Prof. S. M. Shinde
Computer Engineering Department ,
JSCOE & Pune University, India

Abstract— In today's software industries there are many software cost estimation models are there to estimate the financial need to develop a software. The result of the models typically requires obtaining approval to proceed, and factored into business plans, budgets, and other financial planning and tracking mechanisms. Many of the models are providing irrelevant output thereby putting the organization in confusion. So to choose a perfect cost estimation model becomes higher priority for the companies. In our research to rank the cost estimation models proposed system uses previous performance data sets as the evidence. System uses correlation similarity and preference model to identify the rank of the model and thereby cluster the cost estimation models. In our proposed model we have taken many parameters to perform ranking and clustering. In this paper we are demonstrating abilities of software cost estimation method and clustering them; based on their features. It helps us to rank together for further usage of software.

Keywords— cost estimation, correlation, ranking, clustering method.

I. INTRODUCTION

Software cost estimation is tagged part of software development. Software cost estimation [1] has major step in software engineering practice, often determining the success or failure of contract negotiation and project execution. Few of main outcomes of cost estimation are effort, schedule, and staff requirements which are valuable information for project formation and execution.

Prime Component:

- Project bidding and proposal
- Budget and staff allocation
- Project planning, progress monitoring and control
- Investment decision
- Trade off and risk analysis

Due to large number of proposed cost estimation method [1] it is necessary for project manager to systematically base their choice of most accurate model. Time and availability defines approaches of cost estimation. Most of work has focused on algorithmic cost modelling. In this process cost gets analysed using cost estimation model and produce estimated output. One of most used approach is Scott-Knott test. The Scott-Knott test [4] is a multiple comparison procedure based on principles of cluster analysis. The clustering [7] refers to the treatments (methods or in our case models) being compared and not to the individual cases, while the criterion for clustering together gives statistical significance of differences between their mean values.

In this field of research, efforts[8] require to develop a new project is being estimated based on historical data from previous projects. This information can be used by management to improve the planning of personnel, to make more accurate tendering bids, and to evaluate risk factors . Recently, a number of studies evaluating different techniques have been published. The results of these studies revealed mainly three factors:

- There is no standard method to confirm single cost estimation model.
- Old statistical data never narrate present scenarios
- Single parameters based ranking was not enough

These factors have been considered while preparing this proposed work & finding improvement areas. Main Objectives of proposed system are

- First to study the all possible cost estimation models
- Understanding the methods and equations of the model which are using for cost estimation
- Calculating cost Error Precision[3]
- By Using Data sets and similarity measure ranking [8] for complete models and cluster those according to their performance rate e.g. high, medium and low.

II. SYSTEM OVERVIEW

2.1BLOCK DIAGRAM

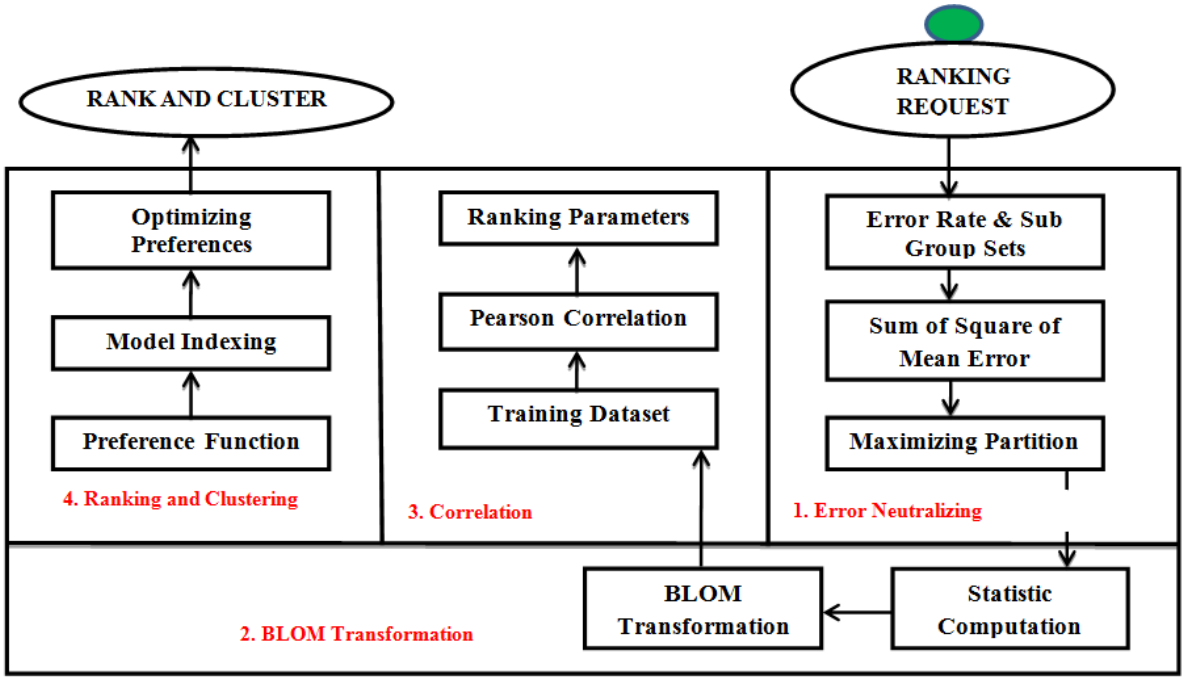

Fig 1.Data flow diagram

2.1.1 Error Neutralization
In this first give ranking request then perform following step
Step 1) Accepting Error Rate from Dataset.
Step 2) Divide mean errors in sub group sets
Step 3) then calculate the group sum of squares of the mean errors
Step 4) Finding the partition that maximizes the value of the sum of square
2.1.2 Bloom transformation

Step 5) Compute the statistics using the following equation

$$\lambda = \frac{\pi G_{j*}}{2(\pi - 2)S^2}$$

Step 6) Applying Bloom transformation

$$\Phi^{-1}\left(\frac{r_i - 3/8}{n + 1/4}\right)$$

It is important to note that the Bloom transformation is monotonous and therefore the order of the values is kept intact. The output of the algorithm is a ranking of the models according to their transformed error measures and, moreover, a clustering scheme where each cluster consists of the sorted models that do not have significant difference in their error measures.

2.1.3 Correlation
Step 7) Accept Accuracy, Opinion Score, Error rate as input
Step 8) detecting similar models with similar attributes using Pearson Rank Correlation using equation (1)
Step 9) Getting Two model similarity
step10) Getting training Dataset
step11) Detecting the similar model in Training dataset
step12) Vector of similar model set
Step 13) Getting All model names

2.1.4 Ranking and clustering
Step15) Detecting users preference over two models using preference function for that use equation 2
Step 16) Detecting Model corresponding order

Step 17) Model indexing
step18) Optimizing Models
Step 19) Ranking models

III. EQUATIONS AND ALGORITHM

3.1 Mathematical equation

1) Pearson Correlation Model Equation

$$r = \frac{\sum xiyi - \frac{\sum xi \sum yi}{10}}{\sqrt{\left(\sum x_i^2 - \frac{(\sum x_i)^2}{10}\right)}\sqrt{\left(\sum y_i^2 - \frac{(\sum y_i)^2}{10}\right)}} \quad \text{..... (Equation 1)}$$

Let x={ Ac, Op, Er} and y=={ Ac, Op, Er}
Take Accuracy, Opinion Score, Error rate as input parameter get rank r as output

2) Preference function

$V^{\Psi}(\rho) = \sum_{i,j:\rho(i)>\rho(j)} \Psi(i,j)$ ..... (Equation 2)

Given a preference function Ψ which assign a score to every pair of model i, j ∈ I, let p=ranking of model in I if and only if such that p(i)>p(j) if and only if i is ranked higher than j in the ranking p. we can define a value function $V^{\Psi}(\rho)$ which measure the consistency of ranking p with preference function in this equation take two different model as input find similarity between them and get cluster of model as output

3.1.2Algorithm for Error Neutralization and Bloom Transformation

Algorithm for Correlational
Input: Dataset D ={e_1,e_2,e_3,….e_n}
Output: Rs as rank
Step 0)start
Step1) Get Set D
Step2)divide D into subgroup
Step3)

$$\sum_e 2 = e_1^2 + e_2^2 + \cdots \ldots \ldots \ldots e_n^2$$

Step4)calculate maximum partition for error rate generate set M_e
Step5) Compare M_e with other subset
Step6) Using Blom transformation get

distribution error rates

$$\Phi^{-1}\left(\frac{r_i - 3/8}{n + 1/4}\right)$$

Step 7) Merge all sub groups
Step8)compute in descending order
Step9)index X rank
Step10)stop

Input: Dataset Ds and training set Ts, Accuracy Ac, Opinion set Op, Error rate Er
Output: Rs as rank
step 1)Accept User Parameter acceptance x,y
step 2) Accept Accuracy, Opinion Score, Error rate
let x={ Ac, Op, Er} and y=={ Ac, Op, Er}
step 3)detecting similar models with similar attributes using Pearson Rank Correlation these are following step for that use equation 2
step i)calculate x*y

$$\sum x{:}\, y = x_1y_1 + x_2y_2 + x_3y_3 + \cdots \ldots \ldots \ldots x_ny_n$$

so $\sum x{:}\, y$=A
step ii)

$$\sum x = x_1 + x_2 + x_3 + \cdots \ldots \ldots \ldots x_n = B$$

Step iii)

$$\sum y = y_1 + y_2 + y_3 + \cdots \ldots \ldots \ldots y_n = C$$

step iv) $N_{r=A-(B*C)/n}$
step v) $\sum_x 2 = x_1^2 + x_2^2 + \cdots \ldots \ldots x_n^2 = M$))
step vi)Q=B^2/n
step vii)Z=$\sqrt{M-Q}$
step viii) $\sum_y 2 = y_1^2 + y_2^2 + y_3^2 + \cdots \ldots \ldots y_n^2$ = V
step ix)u=C^2/n
step x)Z=$\sqrt{V-u}$
step xi)dr=T-Z
step xii) $P_r = nr/dr$

2 Algorithm of Ranking and clustering

1) Getting Two model similarity
2) Getting training Dataset
3) Detecting the similar model in Training dataset
4) Vector of similar model set
5) Getting All model names
6) Detecting users preference over two models using preference function

$$V^{\Psi}(\rho) = \sum_{i,j:\rho(i)>\rho(j)} \Psi(i,j)$$

7) Detecting Model corresponding order
8) Model indexing
9) Optimizing Models
10) Ranking models

IV. CONCLUSION

In our proposed approach we have successfully created 7 to 10 cost estimation software and ask the user to perform the operation of different software as input so that many outcomes of these type of operation can be save in database and considered further as dataset. This dataset gets feed as input of our estimation model where using similarity search, different function, we rank the model and finally cluster them based on rank. Our model can be enhance as web application where different countries cost estimation parameters can be given as input to rank cost estimation model.

ACKNOWLEDGEMENT

Ms Vijaya Wable received her B.E. degree in Information Technology Engineering from SVPM college of engineering malegoan(bk) Pune University, in 2007.Currently she is teaching as Senior lecturer with Department of Computer Engineering at JSPM's Jayawantrao Sawant Polytechnic, Pune since June 2008.In her post graduate course she is doing research work in Ranking and clustering of Software cost estimation models

Prof. Sharmila Shinde . Ph.D. (Pursuing), M. E. She is Working as an Asst. Professor in Computer Engineering Department of J.S.P.M.S Jayawantrao Sawant College of Engineering, Hadapsar Pune-28,

REFERENCES

[1] M. Jorgensen and M. Shepperd "A Systematic Review of Software Development Cost Estimation Studies", vol. 33, no. 1, pp. 33-53, Jan. 2007.
[2] Marian Petre, David Budgen and Jean Scholtz says in "Regression Models of Software Development Effort Estimation Accuracy and Bias". Empirical Software Engineering, 9, 297–314, 2004
[3] B.A.Kitchenham ,L.M.Pickard, S.G.MacDonell band, M.J.Shepperd " What accuracy statistics really measure" , vol. 148, pp. 81-85, June 2001.
[4] S. Lessmann, B. Baesens, C. Mues, and S. Pietsch, "Benchmarking Classification Models for Software Defect Prediction: A Proposed Framework and Novel Findings," , vol. 34, no. 4, pp. 485-496, July/Aug. 2008.,
[5] G. Tsoumakas, L. Angelis, and I. Vlahavas, "Selective Fusion of Heterogeneous Classifiers," Intelligent Data Analysis, vol. 9, no. 6, pp. 511-525, Dec. 2005.
[6] J. Demsar, "Statistical Comparisons of Classifiers over Multiple Data Sets" , J. Machine Learning Research, vol. 7, pp. 1-30, 2006.
[7] S. Lessmann, B. Baesens, C. Mues, and S. Pietsch, "Benchmarking Classification Models for Software Defect Prediction: A Proposed Framework and Novel Findings," IEEE Trans. Software Eng., vol. 34, no. 4, pp. 485-496, July/Aug. 2008.
[8] Nikolaos Mittas and Lefteris Angelis " Ranking and Clustering Software Cost Estimation Models through a Multiple Comparisons Algorithm" , VOL. 39, NO. 4, APRIL 2013

