Multi-criteria decision making approach for the selection of software effort estimation model

Ashu Bansal[a], Brijesh Kumar[a] and Rakesh Garg[b*]

[a]*Department of Computer Science & Engineering, K. R. Mangalam University, Gurgaon, Haryana, India*
[b]*Department of Computer Science & Engineering, Hindu College of Engineering, Sonepat, Haryana, India*

CHRONICLE	ABSTRACT
	Software development with minimum effort has become a challenging task for the software developers. Software effort may be defined as the prediction process of the effort required to develop any software. Many software effort estimation models have been developed in the past, but it is observed that none of them can be applied successfully in all kinds of projects in different environments that raise the problem of the software effort estimation model selection. To select the suitable software effort estimation model, many conflicting selection criteria must be considered in the decision process. The present study emphasizes on the development of a fuzzy multi-criteria decision making approach by integrating Fuzzy Set Theory and Weighted Distance Based Approximation. To show the consistency of the proposed approach, methodology validation is also performed by making comparison with existing methodologies and to check the criticality of the selection criterion, sensitivity analysis is also performed.

1. Introduction

In today scenario, software has grown to be a very significant component for all kinds of systems and software effort estimation is of vital importance for the well balanced management of imminent software. The management, in terms of software effort, is necessary due to the speedy increase in the need of complex software systems at a large scale. Software effort estimation is referred as a prediction process to calculate the most rational amount of efforts required to develop any software based on imperfect and undecided inputs. Effort estimates can be used as input at various stages of the software development process as project plans, budgets, iteration plans, and investment analyses. The prime motive of the software developers is to develop software with better quality by minimizing the required effort. So, it can be stated that the success of any software to some extent relies on the perfect effort estimation. The inaccurate estimation of the required effort for the development of any software will affect adversely to both the customers and developers of the software because it can result in bad quality software, delay in the delivery time, misuse of software and contact cancellation, etc.

During the past decades, a number of models like expert estimation, formal estimation models and combinational estimation models have been developed by the various researchers to portray the software effort estimation. The main center of attention of most of the researchers is based on the creation of new software effort estimation models that results in a new emerging research area as selection of software effort estimation model. Although, the available software effort models are capable of calculating the total effort required to develop any software, but the accuracy differences of various software effort estimation models put forward that no model can be considered as the best one in all the cases. Due to the aforementioned requirements, the researchers focus on an emerging research problem termed as selection of software effort estimation model/technique. A comprehensive systematic review of the past literature enforces that the problem of selection and ranking of software effort estimation models can be represented in the form of multi-criteria decision making problem. The multi-criteria decision making (MCDM) problems are those problems in which a set of existing alternatives are evaluated on a pre-identified set of evaluation criteria and the information about that "which alternative is best" is obtained as the result. The present research argues on the problem "evaluation and selection of software effort estimation models" by modeling it as an MCDM problem. Further, a set of selection criteria is proposed that can contribute in the evaluation process of various software effort estimation models. The rest of the paper is ordered as: section 2 includes the literature review in support to the present problem, the proposed methodology in section 3, Evaluation framework development in section 4 is provided. Section-5 contains model demonstration to show the utility of the proposed methodology followed by the methodology validation and sensitivity analysis in section 6. The results are analyzed in section 7 under major findings and discussions of the paper and section 8 contains conclusion drawn from the present research.

2. Literature Review

An extensive literature survey into two parts was carried out in the context of the present problem of software effort estimation model selection. In the first part, the various models developed by the various researchers were studied with their associated advantages and disadvantages. It is inferred from that the software effort estimation models provided in the past are mainly categorized into two categories namely (i) Algorithmic and (ii) Non-Algorithmic models (Abbas et al. 2012; Pandey, 2013; Ramesh & Karunanidhi 2013). In the second part, the various evaluation metrics that contribute in the evaluation process of the software effort estimation models were analyzed. Leung and Fan provide selection criteria as mean relative error (MRE), mean absolute relative error (MARE), balance relative error (BREbias) to solve the present problem (Leung & Fan, 2002). Moløkken-Østvold et al. (2004) provided mean relative error (MRE), balance relative error bias (BREbias), balance relative error (BRE) and accuracy (ACC) as the selection criteria for the evaluation of software effort estimation models. Tim Menzies et al. (2006) proposed a methodology based on heuristic rejection rules named Coseekmo to rank the various software effort estimation models by considering mean relative error (MRE), mean magnitude of relative error (MMRE) and prediction (PRE) as selection criteria. Basha and Ponnurangam (2010) proposed several evaluation metrics as mean relative error (MRE), mean magnitude of relative error (MMRE), prediction (PRE), root mean square (RMS), relative root mean square (RRMS) for the ranking of software effort estimation models. In the contemporary work, Kaur et al. (2010) used some attributes as mean magnitude of relative error (MMRE), mean square error (MSE), root mean square error (RMSE) and root mean square error (RMSSE) as the selection criteria. Sehra et al. (2013) proposed a model based on Fuzzy Analytic Hierarchy Process by accounting reliability (REL), mean magnitude of relative error (MMRE), Prediction (PRE) and uncertainty (UNC) as selection criteria. Malathi and Sridhar (2012) proposed different selection criteria for the selection and raking of effort estimation models such as prediction (PRED), value accounted for (VAF), variance absolute relative error (VARE), mean absolute relative error (MARE), magnitude of relative error (MRE), root mean square error (RMSE). Wen et al. (2012) provide prediction (PRED), mean magnitude of relative error (MMRE) and median magnitude of relative error (MDMRE) for the purpose of selection and evaluation of models. Garcia-Diaz et al. (2013) developed a methodology based on fuzzy logic for the comparison of two fuzzy logic models for software development effort estimation. Prediction (PRE),

mean error relative (MER), (mean magnitude of error relative) MMER were used as selection criteria in this research. Mittas and Angelis (2013) ranked various effort estimation models by using different criteria as mean absolute error (MAE), magnitude of relative error (MRE) and mean magnitude of relative error (MMRE). Preeth et al. (2014) used three selection criteria for the evaluation of effort estimation model such as magnitude of relative error (MRE), mean magnitude of relative error (MMRE) and prediction (PRED). Nayebi et al. (2015) used different criteria such as prediction (PRED), correlation coefficient (CORR) and Bayesian information correlation (BIC) for the selection of different models. It can be inferred that the problem of selection of software effort estimation model can be represented as a multi-criteria decision making (MCDM) problem. The use of the MCDM approaches to solve the real life problems such as vendor selection (Jarial and Garg, 2012), inventory policy selection (Amit et al., 2014), Software engineering matrices selection (Garg et al., 2013), have been widely accepted in the present time.

3. Proposed Methodology

In the present research, two well-known approaches namely, Fuzzy Set Theory (FST) and Weighted Distance Based Approximation are adopted first time to solve the problem of software effort estimation model selection. FST was first proposed by Zadeh (1965) to deal with the vagueness of the data. A fuzzy set is an ordered pair (F, x) where F is a set and $x : F \rightarrow [0,1]$ i.e. membership function. Fuzzy set theory has already applied in various areas, i.e. scientific environment, control theory, robotics, etc. A triangular fuzzy number (TFN) is usually used to deal with the vagueness of the data which is related to the performance of alternatives with respect to each index. A TFN defined by triplet $T = (u, v, w)$ can be expressed as Eq. (1). The representation of TFN membership function is shown in Fig. 1.

$$\mu_A(x) = \begin{cases} (x-u)/(u-v) & \text{if } u \leq x \leq v, \\ (w-x/(w-v) & \text{if } u \leq x \leq w, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$
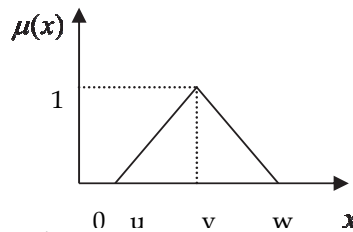

Fig. 1. Triangular fuzzy number membership function

Consider any two triangular fuzzy numbers $T_1 = (u1, v1, w1)$ and $T_2 = (u2, v2, w2)$. Some algebraic operations of the TFNs T_1 and T_2 can be represented as:
Addition, Subtraction and Multiplication '$\oplus$' can be represented as:

$$T_1 \oplus T_2 = (u_1 + u_2, v_1 + v_2, w_1 + w_2) \quad T_1 \oplus T_2 = (u_1 - w_2, v_1 - v_2, w_1 - u_2) \quad T_1 \oplus T_2 = (u_1 u_2, v_1 v_2, w_1 w_2)$$

Weighted Distance Based Approximation (WDBA) measures the distance from the optimal point (best value of alternatives) and non-optimal point (worst value of alternatives). Finally, the alternatives are ranked according to their suitability index. The alternative having the least value of suitability index is ranked at first position and with the maximum value at last. WDBA approach has already been applied in many fields as plant layout design (Rao & Singh, 2012), E-learning websites selection (Jain et al. 2016), COTS component selection (Garg et al., 2016). The steps of WDBA method are given below.

Step 1: Construct the decision rating matrix (D_{ij}), where $i = 1, 2, \ldots\ldots, m, j = 1, 2, \ldots\ldots, n$ represent the performance of ith alternative on the jth index.

Step 2: Calculate the weighted decision rating matrix (W) by using the Eq. (2), where w_j is the weight of jth index.

$$W_{ij} = D_{ij} \times w_j \tag{2}$$

Step 3: Formation of standardized matrix (SM_{ij}) using the Eqs. (3-5), where a_{ij} – adjusted matrix, $\bar{v}_j$ – average value and SD_j – standard deviation.

$$SM_{ij} = \frac{a_{ij} - \bar{v}_j}{SD_j}$$
(3)

$$\bar{v}_j = \frac{1}{n}\sum_{i=1}^{n} a_{ij}$$
(4)

$$SD_j = \left[\frac{1}{n}\sum_{i=1}^{n}(a_{ij} - \bar{v}_j)^2\right]^{1/2}$$
(5)

Step 4: Calculate the distance matrix and composite distance matrix. Finally, the value of suitability index (SI) is obtained by using the Eq. (6).

$$SI = \sqrt{\sum_{j=1}^{n}(CD_{ij})}$$
(6)

4. Development of Evaluation Framework

To meet the main concern of the present research, a hierarchical structured evaluation framework for the evaluation and selection of software effort estimation model (SEEM) is being portrayed based on Fuzzy Weighted Distance based Approximation i.e. FWDBA method considering various evaluation criteria/sub-criteria and their interactions that is capable to work out with the multi-criteria decision making problems in an effective and efficient way.
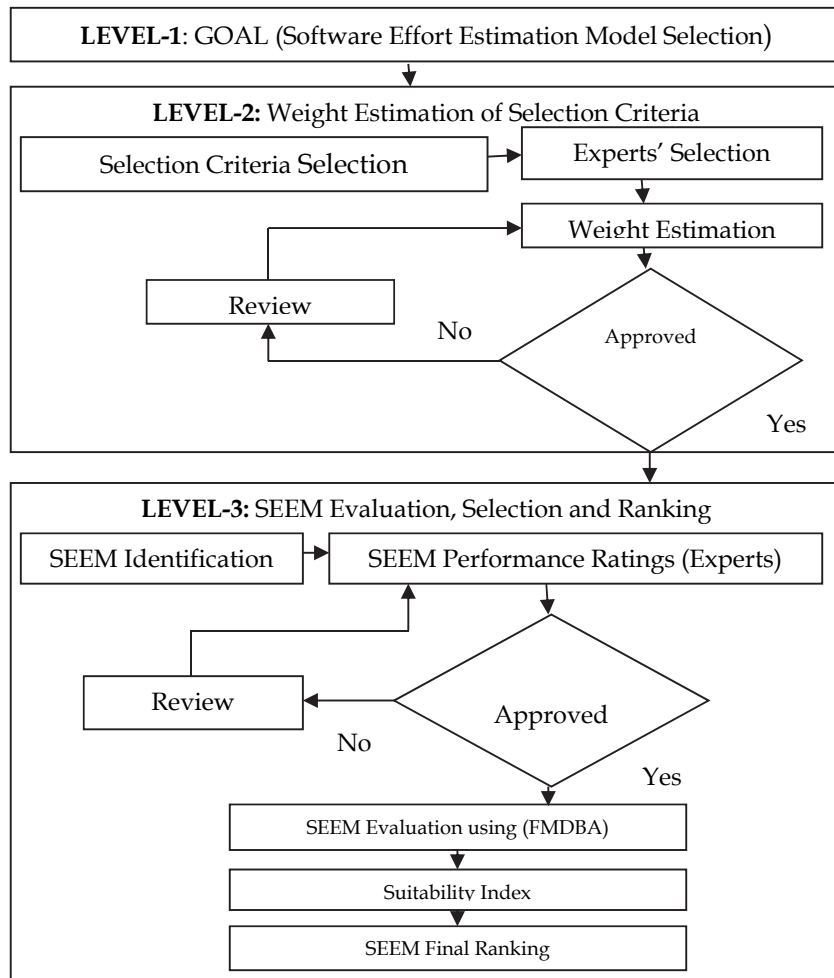

Fig. 2. Development of Hierarchical Evaluation framework

The hierarchy framework has three levels; the first level is a goal level. In the present problem, the goal is to select the best software effort estimation model among several models. The second level contains evaluation criteria against which software effort estimation models are evaluated. Finally, the third level contains the various software effort estimation models being evaluated. The complete hierarchical framework developed in the present research is shown in Fig. 2.

5. Model Demonstration

The step-wise working of the developed evaluation framework for the evaluation and selection of software effort estimation model in the present research follows the three steps as described in next subsections.

5.1. Identification and Selection of Models/Criteria

The first step in the evaluation process involves the identification of various software effort estimation models and the various selection criteria against which the models can be evaluated. After going through an extensive literature survey, total eleven software effort estimation models falling under the category of algorithmic models and total eleven selection criteria were identified and selected for the evaluation purpose. The description of these software effort estimation models and the selection criteria is provided in Table 1.

Table 1
Description of Software Effort Estimation Models and Selection Criteria

Model	Abbreviation	Selection Criteria	Abbreviation
Bailey and Basili Model	M1	Magnitude of Relative Error	C1
Far and Zagorski Model	M2	Root Mean Square	C2
Nelson Model	M3	Prediction	C3
Boeing Model	M4	Root Mean Square Error	C4
Wolverton Model	M5	Mean Absolute Relative Error	C5
Aron Model	M6	Variance Absolute Relative Error	C6
Putnam Model	M7	Value Accounted For	C7
Cocomo Model	M8	Accuracy	C8
Doty Model	M9	Reliability	C9
Walston and Felix Model	M10	Uncertainty	C10
Jensen Model	M11	Mean Absolute Error	C11

5.2. Data collection

Due to lack of maturity in the available secondary data, primary data was collected through the well designed questionnaires. Firstly, a team of five experts (E1, E2, E3, E4, E5) was established deliberately from academia, laboratories as well as IT industries. All the experts have more than 15 years experience in the field of the software effort estimation/software development. Secondly, two questionnaires were designed to collect the data regarding the priority weights of the selection criteria and the performance ratings of the software effort estimation models. The experts were asked to provide this data in the linguistic terms on a 7-point fuzzy scale that is shown in Table 2.

Table 2
Fuzzy Scale (7-Point) Description

Linguistic Terms	Triangular Fuzzy Numbers	Linguistic Terms	Triangular Fuzzy Numbers
Extremely More Important (EMI)	(1,1,1)	Less Important (LI)	(0.2,0.3,0.4))
Very More Important (VMI)	(0.8,0.9,1)	Very Less Important (VLI)	(0.0,0.1,0.2)
More Important (MI)	(0.6,0.7,0.8)	Extremely Less Important (ELI)	(0,0,0)
Important (I)	(0.4,0.5,0.6)		

The data collection in the present research was carried out mainly in two phases as (i) Selection criteria weights data collection and (ii) performance ratings of the models data collection.

5.2.1 Selection criteria weights

The experts provide the data about the priority weights of the selection criteria in the linguistic terms and then the data so obtained was converted in triangular fuzzy numbers and finally a crisp score was

calculated by applying average operation. The data obtained about the priority weights is provided in Table 3.

Table 3
Selection Criteria Weights Calculation

	E1	E2	E3	E4	E5	Aggregate TFNs	Weights
C1	VMI(0.8,0.9,1)	MI(0.6,0.7,0.8)	MI(0.6,0.7,0.8)	VMI(0.8,0.9,1)	MI(0.6,0.7,0.8)	(0.68,0.78,0.88)	0.1099
C2	MI(0.6,0.7,0.8)	I(0.4,0.5,0.6)	MI(0.6,0.7,0.8)	VMI(0.8,0.9,1)	MI(0.6,0.7,0.8)	(0.6,0.7,0.8)	0.0986
C3	I(0.4,0.5,0.6)	LI(0.2,0.3,0.4)	I(0.4,0.5,0.6)	MI(0.6,0.7,0.8)	LI(0.2,0.3,0.4)	(0.36,0.46,0.56)	0.0648
C4	I(0.4,0.5,0.6)	MI(0.6,0.7,0.8)	LI(0.2,0.3,0.4)	I(0.4,0.5,0.6)	MI(0.6,0.7,0.8)	(0.44,0.54,0.64)	0.0761
C5	LI(0.2,0.3,0.4)	I(0.4,0.5,0.6)	MI(0.6,0.7,0.8)	LI(0.2,0.3,0.4)	I(0.4,0.5,0.6)	(0.36,0.46,0.56)	0.0648
C6	MI(0.6,0.7,0.8)	MI(0.6,0.7,0.8)	I(0.4,0.5,0.6)	I(0.4,0.5,0.6)	MI(0.6,0.7,0.8)	(0.52,0.62,0.72)	0.0873
C7	MI(0.6,0.7,0.8)	I(0.4,0.5,0.6)	VMI(0.8,0.9,1)	I(0.4,0.5,0.6)	I(0.4,0.5,0.6)	(0.52,0.62,0.72)	0.0873
C8	VMI(0.8,0.9,1)	VMI(0.8,0.9,1)	MI(0.6,0.7,0.8)	MI(0.6,0.7,0.8)	VMI(0.8,0.9,1)	(0.72,0.82,0.92)	0.1155
C9	LI(0.2,0.3,0.4)	MI(0.6,0.7,0.8)	I(0.4,0.5,0.6)	I(0.4,0.5,0.6)	LI(0.2,0.3,0.4)	(0.36,0.46,0.56)	0.0648
C10	EMI(1,1,1)	EMI(1,1,1)	VMI(0.8,0.9,1)	MI(0.6,0.7,0.8)	I(0.4,0.5,0.6)	(0.76,0.86,0.96)	0.1211
C11	EMI(1,1,1)	MI(0.6,0.7,0.8)	VMI(0.8,0.9,1)	MI(0.6,0.7,0.8)	I(0.4,0.5,0.6)	(0.68,0.78,0.88)	0.1099

5.2.2 Performance ratings of the models

Table 4
Performance Ratings of Software Effort Estimation Models (Linguistic Terms)

Model/Criteria		C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11
	E1	MI	VMI	I	MI	LI	I	MI	MI	LI	I	I
	E2	MI	MI	LI	I	I	MI	VMI	VMI	I	LI	I
M1	E3	I	MI	LI	MI	I	I	MI	MI	I	I	MI
	E4	I	I	I	I	I	I	MI	MI	I	I	MI
	E5	MI	MI	LI	MI	LI	MI	I	VMI	LI	LI	I
	E1	MI	VMI	I	I	LI	MI	MI	VMI	LI	I	EMI
	E2	MI	MI	MI	MI	I	MI	I	VMI	MI	LI	MI
M2	E3	I	MI	I	LI	MI	I	VMI	MI	I	LI	VMI
	E4	MI	VMI	I	I	LI	I	I	MI	I	I	MI
	E5	I	MI	LI	MI	I	MI	I	VMI	LI	LI	I
	E1	MI	MI	VMI	MI	I	MI	EMI	MI	LI	I	MI
	E2	I	VMI	MI	MI	LI	MI	MI	VMI	MI	LI	LI
M3	E3	I	MI	I	LI	MI	MI	VMI	MI	I	I	I
	E4	MI	I	I	I	LI	I	MI	I	LI	I	MI
	E5	LI	MI	LI	MI	I	I	I	VMI	LI	LI	I
	E1	I	MI	VMI	I	LI	VMI	MI	VMI	MI	LI	MI
	E2	LI	MI	MI	MI	LI	EMI	I	VMI	I	LI	LI
M4	E3	I	MI	VMI	LI	I	MI	VMI	MI	I	I	I
	E4	MI	I	I	LI	LI	I	MI	I	MI	I	MI
	E5	LI	MI	LI	MI	I	MI	I	VMI	LI	LI	VMI
	E1	MI	I	MI	VMI	I	MI	VMI	VMI	MI	I	MI
	E2	I	LI	MI	I	MI	MI	I	VMI	I	LI	I
M5	E3	I	I	VMI	VMI	I	MI	MI	MI	I	MI	I
	E4	MI	I	I	LI	LI	I	MI	I	MI	I	MI
	E5	LI	MI	I	MI	I	I	I	MI	I	LI	VMI
	E1	I	MI	VMI	MI	I	VMI	MI	VMI	MI	MI	VMI
	E2	I	LI	MI	I	LI	MI	I	VMI	I	I	I
M6	E3	I	I	VMI	MI	I	MI	MI	MI	MI	MI	I
	E4	MI	MI	I	LI	MI	I	MI	I	MI	I	MI
	E5	LI	MI	MI	MI	I	VMI	LI	MI	I	LI	VMI
	E1	MI	VMI	MI	MI	VMI	EMI	MI	I	VMI	MI	VMI
	E2	MI	I	MI	I	MI	VMI	I	LI	I	I	MI
M7	E3	I	VMI	VMI	MI	VMI	MI	MI	MI	I	MI	VMI
	E4	MI	MI	I	I	MI	I	MI	I	VMI	I	MI
	E5	LI	MI	MI	MI	EMI	VMI	LI	MI	MI	MI	VMI
	E1	VMI	MI	VMI	I	MI	MI	VMI	EMI	I	VMI	MI
	E2	MI	I	VMI	LI	I	I	VMI	VMI	I	I	I
M8	E3	VMI	I	MI	I	MI	VMI	MI	MI	MI	MI	I
	E4	MI	MI	MI	MI	MI	I	MI	I	LI	I	MI
	E5	I	MI	VMI	MI	I	VMI	I	VMI	I	VMI	VMI
	E1	MI	I	VMI	MI	MI	MI	VMI	VMI	MI	VMI	MI
	E2	I	LI	MI	MI	LI	VMI	I	VMI	I	LI	I
M9	E3	I	I	VMI	I	I	MI	EMI	MI	MI	MI	MI
	E4	MI	MI	I	LI	I	I	VMI	I	MI	I	MI
	E5	I	MI	VMI	MI	I	VMI	MI	MI	I	MI	I
	E1	MI	MI	MI	VMI	MI	MI	I	MI	VMI	VMI	I
	E2	MI	LI	MI	MI	I	MI	I	VMI	MI	I	I
M10	E3	I	I	MI	I	I	LI	MI	MI	MI	MI	I
	E4	MI	MI	I	VMI	MI	I	MI	I	MI	I	LI
	E5	I	LI	MI	MI	I	VMI	LI	MI	I	MI	I
	E1	MI	I	MI	VMI	MI	MI	I	VMI	EMI	MI	VMI
	E2	MI	LI	MI	VMI	I	MI	I	VMI	I	I	VMI
M11	E3	I	I	VMI	I	I	MI	MI	MI	MI	MI	I
	E4	MI	I	I	EMI	MI	I	I	VMI	MI	I	MI
	E5	I	MI	I	MI	MI	VMI	LI	MI	VMI	MI	VMI

As described in the above section 5.2.1, the same procedure was adopted to collect the data concerning to the performance rating of the various software effort estimation models against a set of selection criteria. The data so obtained in linguistic terms from the experts is provided in Table 4. Now, the data provided in Table 5 is converted into single numeric value known as crisp value to get the final performance ratings of each software effort estimation model that is provided in Table 5.

Table 5
Performance Ratings of Software Effort Estimation Models

Criteria/Model	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11
C1	0.62	0.7	0.38	0.62	0.42	0.58	0.7	0.78	0.42	0.42	0.58
C2	0.62	0.78	0.5	0.54	0.46	0.62	0.62	0.82	0.46	0.38	0.78
C3	0.54	0.7	0.62	0.58	0.46	0.58	0.78	0.74	0.42	0.42	0.54
C4	0.46	0.66	0.66	0.5	0.38	0.7	0.66	0.78	0.54	0.38	0.62
C5	0.54	0.5	0.66	0.66	0.5	0.62	0.66	0.74	0.58	0.46	0.66
C6	0.5	0.58	0.74	0.58	0.5	0.74	0.58	0.74	0.62	0.54	0.7
C7	0.58	0.74	0.7	0.62	0.86	0.82	0.58	0.54	0.7	0.62	0.82
C8	0.74	0.62	0.82	0.54	0.62	0.7	0.74	0.82	0.5	0.7	0.66
C9	0.58	0.54	0.78	0.58	0.5	0.74	0.82	0.74	0.62	0.62	0.62
C10	0.62	0.5	0.66	0.74	0.58	0.62	0.54	0.7	0.7	0.66	0.46
C11	0.62	0.5	0.66	0.82	0.62	0.66	0.5	0.82	0.78	0.62	0.78

5.3. Evaluation, Selection and Ranking

In the present research, the various software effort estimation models are ranked according to the suitability index value obtained by implementing the proposed methodology i.e. Distance Based Approximation. In the evaluation, selection and ranking procedure, the decision rating matrix is formulated first by taking priority weights of the selection criteria and the performance ratings of the models together. The decision rating matrix so formed is given as:

$$
\begin{bmatrix}
0.62 & 0.70 & 0.38 & 0.62 & 0.42 & 0.58 & 0.70 & 0.78 & 0.42 & 0.42 & 0.58 \\
0.62 & 0.78 & 0.50 & 0.54 & 0.46 & 0.62 & 0.62 & 0.82 & 0.46 & 0.38 & 0.78 \\
0.54 & 0.70 & 0.62 & 0.58 & 0.46 & 0.58 & 0.78 & 0.74 & 0.42 & 0.42 & 0.54 \\
0.46 & 0.66 & 0.66 & 0.50 & 0.38 & 0.70 & 0.66 & 0.78 & 0.54 & 0.38 & 0.62 \\
0.54 & 0.50 & 0.66 & 0.66 & 0.50 & 0.62 & 0.66 & 0.74 & 0.58 & 0.46 & 0.66 \\
0.50 & 0.58 & 0.74 & 0.58 & 0.50 & 0.74 & 0.58 & 0.74 & 0.62 & 0.54 & 0.70 \\
0.58 & 0.74 & 0.70 & 0.62 & 0.86 & 0.82 & 0.58 & 0.54 & 0.70 & 0.62 & 0.82 \\
0.74 & 0.62 & 0.82 & 0.54 & 0.62 & 0.70 & 0.74 & 0.82 & 0.50 & 0.70 & 0.66 \\
0.58 & 0.54 & 0.78 & 0.58 & 0.50 & 0.74 & 0.82 & 0.74 & 0.62 & 0.62 & 0.62 \\
0.62 & 0.50 & 0.66 & 0.74 & 0.58 & 0.62 & 0.54 & 0.70 & 0.70 & 0.66 & 0.46 \\
0.62 & 0.50 & 0.66 & 0.82 & 0.62 & 0.66 & 0.50 & 0.82 & 0.78 & 0.62 & 0.78 \\
0.1099 & 0.0986 & 0.0648 & 0.0761 & 0.0648 & 0.0873 & 0.0873 & 0.1155 & 0.0648 & 0.1211 & 0.1099
\end{bmatrix}
$$

The weighted decision rating matrix can be produced by multiplying the performance ratings of each software effort estimation model with selection criteria priority weights as given in Eq. (2) as:

$$
\begin{bmatrix}
0.0681 & 0.0690 & 0.0246 & 0.0471 & 0.0272 & 0.0506 & 0.0611 & 0.0900 & 0.0272 & 0.0508 & 0.0637 \\
0.0681 & 0.0769 & 0.0323 & 0.0410 & 0.0298 & 0.0541 & 0.0541 & 0.0947 & 0.0298 & 0.0460 & 0.0856 \\
0.0593 & 0.0690 & 0.0401 & 0.0441 & 0.0298 & 0.0506 & 0.0681 & 0.0854 & 0.0272 & 0.0508 & 0.0593 \\
0.0505 & 0.0650 & 0.0427 & 0.0380 & 0.0246 & 0.0611 & 0.0576 & 0.0900 & 0.0349 & 0.0460 & 0.0681 \\
0.0593 & 0.0492 & 0.0427 & 0.0501 & 0.0323 & 0.0541 & 0.0576 & 0.0854 & 0.0375 & 0.0557 & 0.0725 \\
0.0549 & 0.0571 & 0.0479 & 0.0441 & 0.0323 & 0.0646 & 0.0506 & 0.0854 & 0.0401 & 0.0654 & 0.0769 \\
0.0637 & 0.0729 & 0.0453 & 0.0471 & 0.0557 & 0.0716 & 0.0506 & 0.0623 & 0.0453 & 0.0750 & 0.0900 \\
0.0812 & 0.0611 & 0.0531 & 0.0410 & 0.0401 & 0.0611 & 0.0646 & 0.0947 & 0.0323 & 0.0847 & 0.0725 \\
0.0637 & 0.0532 & 0.0505 & 0.0441 & 0.0323 & 0.0646 & 0.0716 & 0.0854 & 0.0401 & 0.0750 & 0.0681 \\
0.0681 & 0.0492 & 0.0427 & 0.0562 & 0.0375 & 0.0541 & 0.0471 & 0.0808 & 0.0453 & 0.0799 & 0.0505 \\
0.0681 & 0.0492 & 0.0427 & 0.0623 & 0.0401 & 0.0576 & 0.0436 & 0.0947 & 0.0505 & 0.0750 & 0.0856
\end{bmatrix}
$$

After the formulation of weighted decision rating matrix, the average and standard deviation values are obtained as 0.0135, 0.0118, 0.0176, 0.0088, 0.0101, 0.0079, 0.0133, 0.0239, 0.0101, 0.0180, 0.01215

and 0.0082, 0.0101, 0.0080, 0.0071, 0.0085, 0.0066, 0.0088, 0.0091, 0.0078, 0.0145, 0.0120 respectively to formulate the standardized matrix. The standardized matrix can be formed by using the Eqs. (3-5) as:

$$
\begin{bmatrix}
0.486 & 0.778 & -2.204 & 0.038 & -0.881 & -1.194 & 0.468 & 0.410 & -1.296 & -0.908 & -0.697 \\
0.486 & 1.556 & -1.234 & -0.814 & -0.578 & -0.669 & 0.913 & 0.913 & -0.964 & -1.241 & 1.129 \\
-0.583 & 0.778 & -0.264 & -0.387 & -0.578 & -1.194 & 1.261 & -0.091 & -1.296 & -0.908 & -1.062 \\
-1.654 & 0.389 & 0.058 & -1.241 & -1.184 & 0.382 & 0.072 & 0.410 & -0.305 & -1.241 & -0.332 \\
-0.583 & -1.167 & 0.058 & 0.465 & -0.275 & -0.669 & 0.072 & -0.091 & 0.030 & -0.575 & 0.033 \\
-1.119 & -0.389 & 0.705 & -0.387 & -0.275 & 0.908 & -0.720 & -0.091 & 0.361 & 0.090 & 0.398 \\
-0.048 & 1.167 & 0.382 & 0.038 & 2.451 & 1.959 & -0.720 & -2.602 & 1.025 & 0.756 & 1.494 \\
2.092 & 0.000 & 1.352 & -0.814 & 0.633 & 0.382 & 0.864 & 0.913 & -0.633 & 1.422 & 0.033 \\
-0.486 & -0.778 & 1.028 & -0.387 & -0.275 & 0.908 & 1.657 & -0.091 & 0.361 & 0.756 & -0.332 \\
0.486 & -1.167 & 0.058 & 1.319 & 0.330 & -0.669 & -1.117 & -0.593 & 1.025 & 1.089 & -1.793 \\
0.486 & -1.167 & 0.058 & 2.172 & 0.633 & -0.143 & -1.513 & 0.913 & 1.688 & 0.756 & 1.129 \\
2.092 & 1.556 & 1.352 & 2.172 & 2.451 & 1.959 & 1.657 & 0.913 & 1.688 & 1.422 & 1.494
\end{bmatrix}
$$

Finally, the composite distance matrix can be produced as:

$$
\begin{bmatrix}
0.0311 & 0.0058 & 0.0531 & 0.0263 & 0.0466 & 0.0758 & 0.0107 & 0.0033 & 0.0374 & 0.0797 & 0.0580 \\
0.0311 & 0.0000 & 0.0280 & 0.0516 & 0.0385 & 0.0526 & 0.0299 & 0.0000 & 0.0295 & 0.1041 & 0.0016 \\
0.0864 & 0.0058 & 0.0109 & 0.0379 & 0.0385 & 0.0758 & 0.0011 & 0.0134 & 0.0374 & 0.0797 & 0.0789 \\
0.1694 & 0.0132 & 0.0070 & 0.0674 & 0.0550 & 0.0189 & 0.0191 & 0.0033 & 0.0162 & 0.1041 & 0.0402 \\
0.0864 & 0.0721 & 0.0070 & 0.0168 & 0.0312 & 0.0526 & 0.0191 & 0.0134 & 0.0115 & 0.0585 & 0.0257 \\
0.1244 & 0.0368 & 0.0017 & 0.0379 & 0.0312 & 0.0084 & 0.0431 & 0.0134 & 0.0073 & 0.0260 & 0.0145 \\
0.0553 & 0.0014 & 0.0039 & 0.0263 & 0.0000 & 0.0000 & 0.0431 & 0.1648 & 0.0018 & 0.0065 & 0.0000 \\
0.0000 & 0.0235 & 0.0000 & 0.0516 & 0.0138 & 0.0189 & 0.0047 & 0.0000 & 0.0226 & 0.000 & 0.0257 \\
0.0553 & 0.0530 & 0.0004 & 0.0379 & 0.0312 & 0.0084 & 0.0000 & 0.0134 & 0.0073 & 0.0065 & 0.0402 \\
0.0311 & 0.0721 & 0.0070 & 0.0042 & 0.0188 & 0.0526 & 0.0587 & 0.0302 & 0.0018 & 0.0016 & 0.1305 \\
0.0311 & 0.0721 & 0.0070 & 0.0000 & 0.0138 & 0.0337 & 0.0766 & 0.0000 & 0.0000 & 0.0065 & 0.0016
\end{bmatrix}
$$

Finally, the value of suitability index (SI) is obtained using equation (6), and the final ranking of eleven software effort estimation models based on their suitability index is shown in Table 6 and Fig. 3.

Table 6
Rankings of Software Effort Estimation Models using F-WDBA Method

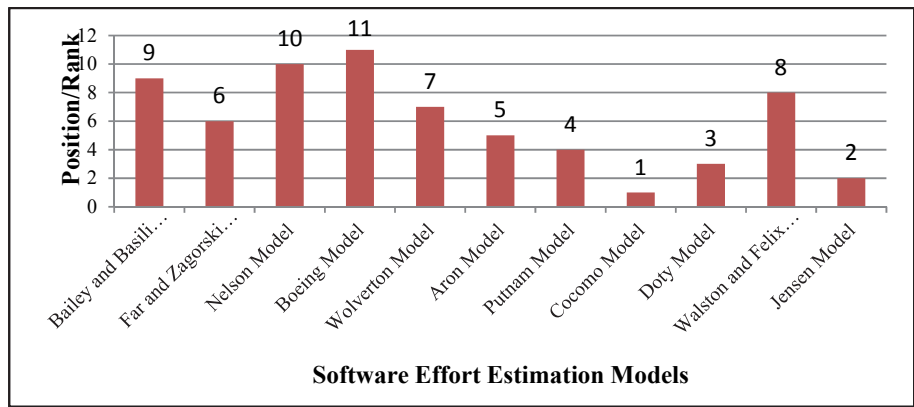| Model	Suitability Index	Position/Rank	Model	Suitability Index	Position/Rank
M1	0.654392	9	M7	0.550878	4
M2	0.606051	6	M8	0.401528	1
M3	0.682905	10	M9	0.503969	3
M4	0.71773	11	M10	0.639588	8
M5	0.628417	7	M11	0.492655	2
M6	0.587477	5			

Fig. 3. Rankings of Software Effort Estimation Models obtained from F-WDBA

6 Methodology Validation and Sensitivity Analysis

To validate the results obtained from proposed approach i.e. F-WDBA, the present software effort estimation model selection problem is also solved using the same dataset presented in illustrated example is solved using two well-known MCDM approaches namely, AHP and TOPSIS. The comparative ranking statistics of the software effort estimation models abbreviated as (M1-M11) based on eleven selection criteria (C1-C11) using AHP, TOPSIS and F-WDBA along-with ranking differences are given in Table 7 and the comparison of the ranking obtained from three approaches as WDBA, AHP and TOPSIS are represented in Fig. 4.

Table 7
Comparison of the rankings obtained from AHP, TOPSIS and F-WDBA method

Models	Rank: AHP (A)	Rank: TOPSIS (B)	Rank: F-WDBA (C)	Ranking Differences		
				(D=A-B)	(D=B-C)	(D=A-C)
M1	10	9	9	1	0	1
M2	6	7	6	-1	1	0
M3	9	10	10	-1	0	-1
M4	11	11	11	0	0	0
M5	8	8	7	0	1	1
M6	5	6	5	-1	1	0
M7	2	2	4	0	-2	-2
M8	1	1	1	0	0	0
M9	4	4	3	0	1	1
M10	7	5	8	2	-3	-1
M11	3	3	2	0	1	1

Further, Spearman's rank correlation test is also performed to test the existence of relationship between the rankings obtained from F-WDBA, AHP and TOPSIS. The test statistics for the Spearman's rank correlation test are given in Table 8. According to Spearman's test, any value of Spearman's Rank greater than 0.6 is considered good and shows the positive significant relationship between the results. It is observed that the value of Spearman's rank obtained in present work is 0.955 and 0.918 that affirms the strong positive relationship between the rankings obtained from WDBA, AHP and TOPSIS.

Table 8
Spearman's Rank Correlation Test Statistics

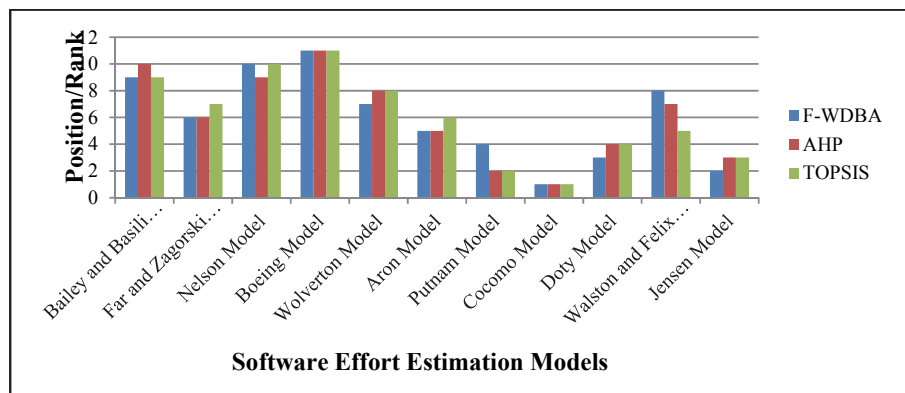| Set of Ranking Methods	(A-C)	(B-C)
Squared Sum $\left(\sum D^2\right)$	10	18
Spearman's Rank - Correlation Coefficient (r_s)	0.955	0.918

Fig. 4. Comparative Rankings obtained from F-WDBA, AHP and TOPSIS

In order to check the criticality of the selection criteria, one-way sensitivity analysis is also carried out in the present research. The criticality of selection criteria here means that which selection criteria is most important to which alternative i.e. software effort estimation model and which one is least important for the alternatives. To perform the sensitivity analysis in the present work, the performance rating of each software effort estimation model is increased by 10% for each selection criteria one by one and the impact of these changes in the performance ratings on the suitability index of each software effort estimation model is analyzed that is represented in Figure 5 that provides very useful information about the criticality of the selection criteria for each software effort estimation model.
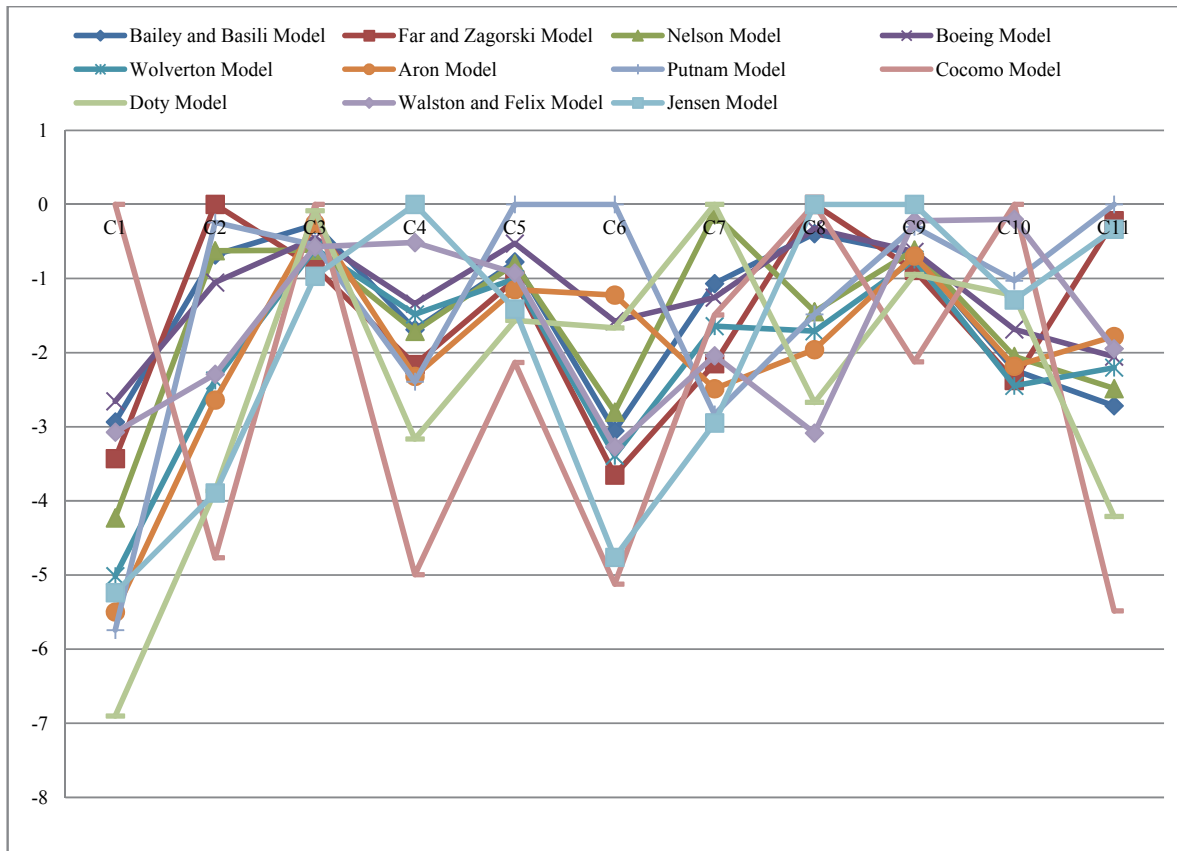

Fig. 5. Sensitivity Analysis Statistics of Software Effort Estimation Models

7. Major findings and discussions

The present research emphasizes on the evaluation and selection of eleven software effort estimation models based on eleven selection criterion by implementing a well known MCDM approach namely, F-WDBA. Based on the results so obtained, some major findings of the research were discussed as given below.

In the present research, Fuzzy Weighted Distance Based Approach i.e. F-WDBA is applied to solve the problem software effort estimation model selection. According to proposed approach, the software effort estimation having the smallest value of suitability index is ranked at top position i.e. number-1 and model with largest value of suitability index is ranked at last position. It is depicted in Table 3 and Fig. 2 that Cocomo Model abbreviated as (M8) is ranked at top i.e. number-1 having least value of suitability index as 0.401528 followed by Jenson Model (M11) at number-2 with 0.492655 whereas Boeing Model (M4) is ranked at last position i.e. number-11 having the largest suitability index as 0.717730. To validate the results obtained by implementing proposed approach in the present research, a comparison is made with the existing MCDM approaches as AHP and TOPSIS. Further, Spearman's Rank Correlation test is carried out that depicts a strong positive relationship between the rankings obtained from

WDBA, AHP and TOPSIS. A modest attempt is also made to find the criticality of the selection criteria by performing one-way sensitivity analysis. The statistics of the sensitivity analysis are provided in Figure 5 that depicts that Prediction (C3) is the least critical selection criteria for many software effort estimation models as Bailey and Basili Model (M1), Wolverton Model (M5), Aron model (M6) and Cocomo Model (M8) whereas Magnitude of Relative Error (C1) is found most critical one Nelson Model (M3), Boeing Model (M4), Wolverton Model (M5), Aron model (M6), Putnam Model (M7), Doty Model (M9) and Jensen Model (M11). The results obtained from the present research concerning with the rankings of software effort estimation models obtained from proposed methodology i.e. F-WDBA and the selection criteria criticality for each model are further compiled in Table 9 as given below.

Table 9
Compiled Results of the Present Research

Model Name	Ranking (WDBA)	Selection Criteria Criticality	
		Least Critical	**Most Critical**
Bailey and Basili Model	9	Prediction	Variance Absolute Relative Error
Far and Zagorski Model	6	Root Mean Square, Accuracy	Variance Absolute Relative Error
Nelson Model	10	Value Accounted For	Magnitude of Relative Error
Boeing Model	11	Accuracy	Magnitude of Relative Error
Wolverton Model	7	Prediction	Magnitude of Relative Error
Aron Model	5	Prediction	Magnitude of Relative Error
Putnam Model	4	Mean Absolute Relative Error, Variance Absolute Relative Error, Mean Absolute Error	Magnitude of Relative Error
Cocomo Model	1	Prediction, Magnitude of Relative Error, Accuracy, Mean Absolute Error	Mean Absolute Error
Doty Model	3	Value Accounted For	Magnitude of Relative Error
Walston and Felix Model	8	Variance Absolute Relative Error	Uncertainty
Jensen Model	2	Root Mean Square Error, Reliability	Magnitude of Relative Error

8 Conclusion

The present study analyses the forthcoming issue of the selection of software effort estimation models related to the IT sector. The present problem is modeled as MCDM problem and novel simplified and well run approach, namely weighted distance based approximation is proposed to solve it. WDBA is more effectual as compare to other existing MCDM approaches due to the involvement of uncomplicated matrix operations. The weights of the identified selection criteria are also incorporated in the present research that can play a very significant role in the evaluation process of software effort estimation models. No pairwise comparison is done to find the weights of selection criteria as compare to other approaches like AHP. For example, as in the present study, eleven selection criteria are considered, so, total 121 comparisons are required to calculate the selection criteria weights in case of pairwise comparison based approaches. Although, the present approach i.e. WDBA has great advantages such as less complexity, easy to implement, some limitations are also associated as development of hierarchical model needed if selection criteria are more. Further, the concept of methodology validation performed in this research strengthen the proposed approach i.e. WDBA. Sensitivity analysis carried out here is very important and crucial because it can help a lot to the software developers as well as researchers to make new effective and efficient model. The present research can be further enhanced by performing two way sensitivity analysis to check the criticality of the various selection criteria for each software effort estimation model and developing any computerized system capable to solve the present MCDM problem i.e. software effort estimation models evaluation, selection and ranking.

References

Abbas, S. A., Liao, X., Rehman, A. U., Azam, A., & Abdullah, M. I. (2012). Cost estimation: A survey of well-known historic cost estimation techniques. *Journal of Emerging Trends in Computing and Information Sciences*, *3*(4), 612-636.

Amit, G., Ramesh, K., & Tewari, P. C. (2014). Ranking of inventory policies using distance based approach method. *World Academy of Science, Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering, 8*(2), 395-400.

Basha, S., & Ponnurangam, D. (2010). Analysis of empirical software effort estimation models. *arXiv preprint arXiv:1004.1239*.

Garcia-Diaz, N., Lopez-Martin, C., & Chavoya, A. (2013). A comparative study of two fuzzy logic models for software development effort estimation. *Procedia Technology*, 7, 305-314.

Garg, R. K., Sharma, K., Nagpal, C. K., Garg, R., Garg, R., & Kumar, R. (2013). Ranking of software engineering metrics by fuzzy-based matrix methodology. *Software Testing, Verification and Reliability, 23*(2), 149-168.

Garg, R., Sharma, R., & Sharma, K. (2016). Ranking and selection of commercial off-the-shelf using fuzzy distance based approach. *Decision Science Letters, 5*(2), 201-210.

Jain, D., Garg, R., Bansal, A., & Saini, K. K. (2016). Selection and ranking of E-learning websites using weighted distance-based approximation. *Journal of Computers in Education*, 3(2), 193-207.

Jarial, S. K., & Garg, R. K. (2012). Ranking of vendors based on criteria by MCDM-matrix method-a case study for commercial vehicles in an industry. *International Journal of Latest Research in Science & Technology, 1*(4), 337-341.

Kaur, J., Singh, S., Kahlon, K. S., & Bassi, P. (2010). Neural network-a novel technique for software effort estimation. *International Journal of Computer Theory and Engineering*, 2(1), 17.

Leung, H., & Fan, Z. (2002). Software cost estimation. *Handbook of Software Engineering*, Hong Kong Polytechnic University, 1-14.

Malathi, S., & Sridhar, S. (2012). Analysis of size metrics and effort performance criterion in software cost estimation. *Indian Journal of Computer Science and Engineering, 3*(1), 24-31.

Menzies, T., Chen, Z., Hihn, J., & Lum, K. (2006). Selecting best practices for effort estimation. *IEEE Transactions on Software Engineering, 32*(11), 883-895.

Mittas, N., & Angelis, L. (2013). Ranking and clustering software cost estimation models through a multiple comparisons algorithm. *IEEE Transactions on Software Engineering, 39*(4), 537-551.

Moløkken-Østvold, K., Jørgensen, M., Tanilkan, S. S., Gallis, H., Lien, A. C., & Hove, S. W. (2004, September). A survey on software estimation in the Norwegian industry. *In Software Metrics, 2004. Proceedings. 10th International Symposium on (pp. 208-219). IEE*E.

Nayebi, F., Abran, A., & Desharnais, J. M. (2015). Automated selection of a software effort estimation model based on accuracy and uncertainty. *Artificial Intelligence Research, 4*(2), p45.

Pandey, P. (2013, April). Analysis of the techniques for software cost estimation. *In Advanced Computing and Communication Technologies (ACCT), 3rd International Conference on (pp. 16-19)*.

Preeth, R., ShivaKumar, N., & Balaji, N. (2014). Software effort estimation using attribute refinement based adaptive Neuro Fuzzy Model. *International Journal of Innovative Research in Science Engineering and Technology, 3*(3).

Ramesh, K., & Karunanidhi, P. (2013). Literature survey on algorithmic and non-algorithmic models for software development effort estimation. *International Journal of Engineering And Computer Science* ISSN, 2319-7242.

Rao, R. (2012). Weighted Euclidean distance based approach as a multiple attribute decision making method for plant or facility layout design selection. *International Journal of Industrial Engineering Computations, 3*(3), 365-382.

Sehra, S. K., Brar, D., Singh, Y., & Kaur, D. (2013). Multi criteria decision making approach for selecting effort estimation model. *arXiv preprint arXiv:1310.5220*.

Wen, J., Li, S., Lin, Z., Hu, Y., & Huang, C. (2012). Systematic literature review of machine learning based software development effort estimation models. *Information and Software Technology, 54*(1), 41-59.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control, 8*, 338–353.