A TOOL FOR SELECTING SUITABLE SOFTWARE PROJECT EFFORT ESTIMATION MODEL AT EARLY PHASES

ERKEN AŞAMADA YAZILIM PROJESİ EFOR KESTİRİMİ İÇİN UYGUN MODEL SEÇİM ARACI

DUYGU DENİZ ERHAN

ASSOC. PROF. DR. AYÇA KOLUKISA TARHAN

Supervisor

Submitted to

Graduate School of Science and Engineering of Hacettepe University

as a Partial Fulfillment to the Requirements

for the Award of the Degree of Master of Science

in Computer Engineering

2021

To my lovely son, Kerem

and dear husband, Musa...

ABSTRACT

A TOOL FOR SELECTING SUITABLE SOFTWARE PROJECT EFFORT ESTIMATION MODEL AT EARLY PHASES

Duygu DENİZ ERHAN

Master of Science, Department of Computer Engineering

Supervisor: Assoc. Prof. Dr. Ayça KOLUKISA TARHAN

September 2021, 104 pages

Effort estimation is one of the important factors affecting the success of software projects. In order to support this, many effort estimation methods have been developed from past to present. The reliability of the effort estimation of a project depends on the choice of the most appropriate method for the project characteristics and the estimation context. Even if a good performing method is used, the estimation results may remain to be inaccurate if an appropriate estimation method is not selected as appropriate to the project context. In this study, we proposed a tool for selecting the most suitable estimation method for a software project by considering the project characteristics and the stakeholder needs. To do this, first, an expert-opinion survey was prepared based on the key features of the commonly used estimation methods that have been frequently referred to in literature. The expert-opinion survey was answered by experts who carried out scientific studies in the field of software effort estimation. Then, a questionnaire was built for eliciting information about project characteristics from an estimator who wants to carry out effort estimation for his/her project. In this phase, firstly, a decision matrix was created in the light of experts' opinions. With the decision matrix, the estimator can select the most suitable method for his/her estimation by answering the questionnaire. Secondly, another approach was created as a decision mechanism. The decision mechanism has two steps. First, prepared decision tree is run and second, multi-criteria decision analysis (MCDA) methodologies are used among the models that are the result of the first elimination with the estimator’s opinions. A tool was developed for the simpler use of this approach. Accordingly, estimator is provided to select the best-fit method using the tool without needing to know the calculation details of the selection. The tool proposes the most appropriate method by first following the decision tree mechanism and then calculating the method ranks. To investigate the validity of the proposed approach, sample studies were conducted and the questionnaire was answered using the ISBSG dataset. Also, we prepared a multiple-case study for the validation of the approach proposed. At the end of the study, the appropriateness of the proposed approach was discussed.

Keywords: Effort Estimation, Software Effort, Estimation Method, Method Selection, Decision Matrix, Decision Tree, Fuzzy TOPSIS, Expert Opinion.

ÖZET

ERKEN AŞAMADA YAZILIM PROJESİ EFOR KESTİRİMİ İÇİN UYGUN MODEL SEÇİM ARACI

Duygu DENİZ ERHAN

Yüksek Lisans, Bilgisayar Mühendisliği

Tez Danışmanı: Doç. Dr. Ayça KOLUKISA TARHAN

Eylül 2021, 104 sayfa

Efor tahmini, yazılım projelerinin başarısını etkileyen önemli faktörlerden biridir. Bunu desteklemek için geçmişten günümüze pek çok efor tahmin yöntemi geliştirilmiştir. Bir projenin efor tahmininin güvenilirliği, proje özellikleri ve tahmin bağlamı için en uygun yöntemin seçimine bağlıdır. İyi performans gösteren bir yöntem kullanılsa bile, proje kapsamında uygun bir tahmin yöntemi seçilmezse tahmin sonuçları hatalı kalabilir. Bu çalışmada, proje özelliklerini ve paydaş ihtiyaçlarını dikkate alarak bir yazılım projesi için, en uygun tahmin yöntemini seçmek için bir araç önerildi. Bunun için öncelikle, literatürde sıklıkla atıfta bulunulan ve yaygın olarak kullanılan tahmin yöntemlerinin temel özelliklerine dayalı bir uzman görüşü anketi hazırlandı. Uzman görüşü anketi, yazılım efor tahmini alanında bilimsel çalışmalar yapan uzmanlar tarafından cevaplandı. Ardından, projesi için efor tahmini yapmak isteyen bir tahmin ediciden, proje özellikleri hakkında bilgi almak için bir anket oluşturuldu. Bu aşamada ilk olarak, uzman görüşleri ışığında bir karar matrisi oluşturuldu. Karar matrisi ile tahminci, anketi cevaplayarak tahmini için en uygun yöntemi seçebilmektedir. İkinci olarak, karar mekanizması olarak başka bir yaklaşım oluşturuldu. Karar mekanizmasının iki adımı vardır. İlk adımda, hazırlanan karar ağacı çalıştırılır ve ikinci adımda, tahmin edici görüşleri ile ilk eleme sonucu olan modeller arasında çok kriterli karar analizi (MCDA) metodolojileri kullanılır. Bu yaklaşımın kullanımını basitleştirmek için bir yazılım aracı hazırlanmıştır. Böylelikle tahminleyicinin, seçimin hesaplama detaylarını bilmesine gerek olmadan, aracı kullanarak en uygun yöntemi seçebilmesi sağlanmıştır. Araç kendi işleyişi içinde, önce karar ağacı mekanizmasını takip edip ardından, yöntemlerin sırasını hesaplayarak en uygun yöntemi önermektedir. Önerilen yaklaşımın geçerliliğini sınamak için örnek çalışmalar yapılmış ve anket, ISBSG veri seti kullanılarak cevaplanmıştır. Ayrıca, doğrulama için çoklu-vaka çalışması hazırlanmıştır. Çalışma sonunda, önerilen yaklaşımın uygunluğu tartışılmıştır.

Anahtar Kelimeler: Efor Tahmini, Yazılım Eforu, Tahmin Yöntemi, Yöntem Seçimi, Karar Matrisi, Karar Ağacı, Bulanık Hesaplama, Uzman Görüşü.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude and thanks to my supervisor Assoc. Prof. Dr. Ayça KOLUKISA TARHAN. I will always appreciate all she has done. Her expert guidance, supportive and positive approach, and patience throughout this period had a great impact on the completion of the work.

I would like to thank my thesis committee members Prof. Dr. Mehmet Önder EFE, Assoc. Prof. Dr. Aysu BETİN CAN, Assoc. Prof. Dr. Murat AYDOS, Asst. Prof. Dr. Tülin ERÇELEBİ AYYILDIZ for their time, useful comments and suggestions.

I am deeply grateful to my parents for their love, patience and support throughout my life. Also, I thank them for encouraging me to believe in myself.

I also want to thank my wonderful son, Kerem, for his innocent and beautiful heart, for his patience for the times I could not spend with him. Thanks to him, he made me discover my ability to love endlessly.

I have no words to express my gratitude to my dear husband Musa for him unlimited support, patience and encouragement. I would like to thank my husband with all my heart, who I know I would never have succeeded without him, who has made my life beautiful and enabled me to succeed, and who has always been patient. I am so grateful for him being in my life.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AB	Analogy Based
AHP	Analytic Hierarchical Process
BN	Bayesian Networks
CBR	Case-Base Reasoning
COCOMO	Constructive Cost Model
CRR	Correlation Coefficient
DT	Decision Trees
EJ	Expert Judgement
ELECTRE	Elimination and Choice Expressing the Reality
ISBSG	International Software Benchmarking Standards Group
LR	Linear Regression
MAE	Mean Absolute Error
MARE	Mean Absolute Relative Error
MCDA	Multiple Criteria Decision Analysis
ML	Machine Learning
MMRE	Mean Magnitude of Relative Error
MRE	Magnitude of Relative Error
NIS	Negative Ideal Solution
NN	Neural Networks
PIS	Positive Ideal Solution
PROMETHEE	The Preference Ranking Organization METHod for Enrichment of Evaluations
SEE	Software Effort Estimation
SVR	Support Vector Regression
TOPSIS	Technique for Order Performance by Similarity to Ideal Solution

1. INTRODUCTION

Software effort estimation (SEE) is the process of predicting the amount of effort required to build a software system. For effective planning, effort and schedule estimation is required for a project. In order to provide this benefit, estimation process must be accurate and reliable but this is a difficult task. In order to address this problem, many estimation methods have been proposed by researchers and many of the proposed methods have been shown to give successful results.

Nevertheless, there is no estimation method that makes the most accurate estimates in all projects [1][2]. Estimation methods make successful estimations for projects that provide certain characteristics (organizational structure, type of project, development environment etc.). It is stated that the most successful effort estimation method can change for a given dataset because different criteria are used [3]. Since the estimation method that gives accurate results will change even for different projects within the same organization, the selection of methods on the basis of the organization may not give correct results. Although methods focusing on specific project features are continually proposed for more accurate estimates in literature [4][5], it is necessary to perform analyses regarding the attributes of method, project and environment each time and even to use expert knowledge for determining which method is suitable. An accurate effort estimation can be achieved only by selecting an estimation method best matched to the estimation context.

Several studies on selecting the suitable estimation method have been proposed by examining the project properties and datasets to be used [4][5]. Although these studies have shown that the success of estimation methods can change according to the project characteristics and dataset, they do not propose a general method for different types of projects and environments. The existing selection methods are not feasible for a new project. Since the method is chosen according to the mean magnitude of relative error value of the estimations in old projects, the characteristics of the new project are not considered, so it may not be suitable for the new project.

In this study, project characteristics that affect the success of estimation methods were examined. For this purpose, an expert-opinion survey was prepared and the relationship between the project characteristics and estimation methods was studied. The survey was realized by referring to the knowledge of the experts having published scientific studies on software effort estimation. It was aimed to determine the most suitable estimation method for given project characteristics and stakeholder needs by using the data obtained from the expert-opinion survey and answers from estimator questions. Multiple Criteria Decision Analysis (MCDA) method was used while processing the data from the expert-opinion survey into a decision matrix. Then, in an example estimation scenario based on ISBSG dataset, a user was asked to answer a set of estimator questions prepared, and the most suitable estimation method was selected with the information of a new project to be estimated. In order to take the study one step further, a two-phase decision approach was studied with expert opinion data. In this approach, first, the decision tree was prepared with the data. After the preliminary elimination in the methods with the selection questions in the tree, the most appropriate selection was made with the Fuzzy TOPSIS calculation. The process steps followed to create the approach proposed is provided in Figure 1.1.

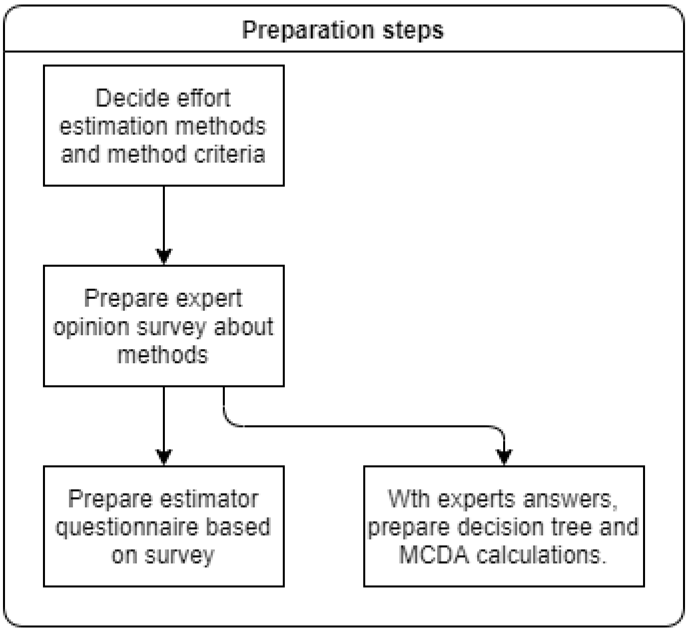


Figure 1.1. Process Steps Followed to Create the Approach

The rest of this thesis is organized as follows. Section 2 provides background and related work on software effort estimation, method selection, and MCDA. Section 3 explains the proposed evaluation approaches, alternative estimation methods used in this study, and method selection criteria. In Section 4, it is exemplified how decision matrix values are formed over the answers to expert-opinion survey questions as well as how decision tree and Fuzzy TOPSIS are prepared and how they work. Section 5 presents example evaluation using ISBSG dataset and related estimation assumptions, and explains the feasibility of the proposal. Also in the same section, with a multiple-case study application, the validity of the decision analysis approach is investigated. Finally, Section 6 discusses the weaknesses in the proposed approach, concludes the thesis with a summary of this study and plans for future work.

2. BACKGROUND AND RELATED WORK

2.1. Software Effort Estimation and Method Selection

Since effort estimation method selection is a major factor in estimation, there are many studies in literature that examine and classify methods. At the same time, due to the importance of the criteria that affect the decision in method selection, there are many studies that investigate the criteria of various parameters that affect the methods. We chose the methods and the criteria we used to prepare the expert-opinion survey based on the studies that we describe below.

Wen et al. [6] made a systematic literature review on machine learning (ML) based software development effort estimation models. They analyzed ML based models from four aspects: type of ML technique, estimation accuracy, method comparison, and estimation context. After reviews, the authors found that eight types of ML techniques are mostly used, including Case-Based Reasoning, Artificial Neural Networks, Decision Trees, Bayesian Networks, Support Vector Regression, Genetic Algorithms, Genetic Programming, Association Rules. They suggested that ML methods are usually more accurate than non-ML methods. Also, they listed the strengths and weaknesses of the ML techniques used in software effort estimation. This study has been decisive in the selection of the ML methods that we use in this study. It also helped us to define criteria in our survey by listing the strengths and weaknesses of the relevant methods.

Jorgenson et al. [7] prepared a basis for the studies about improvement of software development cost estimation. They explored the question “What are the most investigated estimation methods and how has this changed over time?” and as a result, they showed the distribution of articles on different estimation approaches per period and in total. Also, they made recommendations for future estimation researches.

Marco et al. [8] made a systematic review on software effort estimation methods and reported that the number of studies on the subject is increasing. They also pre pared a list of the best performing methods and the most used methods. The most active and influential researchers were also shown in their paper. We invited these researchers to answer our expert-opinion survey.

Idri et al. [9] analyzed analogy-based SEE techniques according to criteria and the studies from some perspectives (estimation context, accuracy comparison, estimation accuracy etc.) and they found that more estimation techniques should be developed. They also said that accuracy in effort estimation depends on several categories of parameters. These parameters are: Dataset characteristics used (size, missing value etc.), analogy process configuration (adaptation formula, feature selection etc.), evaluation method used (n-fold cross validation, disagreement etc.). This study has been a guide in determining some of our method selection criteria.

Bilgaiyan et al. [10] made a review on software cost estimation in agile software development. They prepared a study in which different estimation methods are required to be successful, and discussed the difficulties of the methods.

Shekhar et al. [11] made a comprehensive review on software effort estimation methods. They explained the working principles of many methods. In addition, by listing the advantages and disadvantages of the methods, they shed light on the desired and to be avoided situations in the use of the methods.

Chirra et al. [12] tabulated all the methods in software cost estimation based on their type, amount of data required, validation methods used by them, weaknesses and strengths. They discussed the detailed results about the methods from several perspectives, including: type (algorithmic method, learning oriented method etc.), strengths, weakness, accuracy, data (limited, extensive etc.), and validation (cross validation method, Jackknife method etc.).

In summary, we used the studies mentioned so far in the selection of methods and criteria. We received support in the selection of ML methods from the study by Wen et al. [6], which lists the most used ML techniques and presents the strengths and weaknesses of these techniques. This study, which prepared the best performing methods and the most used methods lists, also guided for method selection. In addition, this study published a list of the most active and influential researchers, helping to identify the experts we would invite to our survey [8]. We also worked on determining the parameters with the study Idri et al. [9], which says that accuracy in effort estimation depends on some parameters. In addition, the studies discussing the difficulties of the methods [10] and listing the advantages and disadvantages of the methods [11] were also used in the selection of methods and criteria. Also, the study by Chirra and Reza [12], which tabulates and discusses all methods from many aspects, was used in the preparation of the survey.

In addition to the studies summarized above, there are studies that associate SEE method selection with various criteria and want to structure it. We also overview these studies below.

In 2012, Sehra et al. [4] proposed a method for selecting an effort estimation method based on the environment and the project type by using Fuzzy Analytic Hierarchy Process. They used reliability, mean magnitude of relative error (MMRE), prediction (Pred), and uncertainty criteria as input for their method. Their selected decision alternatives are Expert Judgement, COCOMO, and Fuzzy Neural Network based effort estimation methods.

In 2017, Bansal et al. [5] proposed fuzzy weighted distance-based approximation (WDBA) to solve selecting an effort estimation method problem based on MCDA. They found that WDBA is more effective than other MCDA solutions due to the lack of complex matrix operations. They used magnitude of relative error (MRE), root mean square (RMS), prediction (Pred), root mean square error (RMSE), mean absolute relative error (MARE), variance absolute relative error (VARE), value accounted for (VAF), accuracy, reliability, uncertainty, and mean absolute error (MAE) as input to their method. They selected eleven algorithmic effort estimation methods as decision alternatives.

In 2015, Nayebi et al. [3] proposed an approach for selecting a machine learning effort estimation method for specific datasets. They selected nine machine learning methods as decision alternatives. They used prediction, correlation coefficient (CRR), and Bayesian Information Criterion (BIC) as inputs to their approach. They compared SEE methods based on these criteria by evaluating nine different datasets.

Ozakinci and Tarhan [13] aimed to identify software defect prediction methods in the early stages of the project, which would give the most accurate result in defect prediction. In this study, the authors determined the criteria based on the project, data, and method features considering the related studies in literature. Then, they sent a survey to the experts and asked them to evaluate the criteria against the prediction methods. At the end, using the MCDA tool, they prepared a questionnaire for users to choose the appropriate method for software defect prediction. Our study employed a similar approach as specific to software effort estimation.

As a result, using the project type and environmental factors in the selection of the effort estimation method, the study [4] received four inputs with MMRE values and it chose between EJ, COCOMO and Fuzzy Neural Network methods. It differs from our study in that its inputs have method properties and it only has three methods to choose from. Bansal et al. [5] worked only on algorithmic methods in method selection and proposed a fuzzy weighted distance-based approach (WDBA). They used method properties such as MRE and MARE as inputs in their studies. Their study differs from our work by not taking user input and using methods with a certain classification. In the study by Nayebi et al. [3], only ML methods were studied on certain datasets in the method selection. The authors compared SEE methods by evaluating nine different datasets with inputs such as the correlation coefficient. In this study [3], inputs were not user-based but method-based, and a selection was made according to methods' performances on certain datasets. It differs from our study because of its scope constancy. Ozakinci and Tarhan [13] have done their work in the field of defect prediction. The methods and criteria chosen are different due to the fact that our study is in a different field. Therefore, the prepared questionnaire, the experts reached and the answers of the experts on which we base our study are also different. Thus, all of the studies used in the decision stages are different studies that follow a similar path. In our study, we have worked from the beginning by applying the steps followed by the study [13] into our own context. This study has guided us in collecting data and how to use it.

2.2. Decision Methods

2.2.1. Decision Trees

Decision tree is one of the most used techniques in decision-making mechanisms. Decision trees help to see the big picture of a particular problem [14]. In this method, the solution is reached by considering the decisions in the tree nodes with a "top-down" approach [15]. The first node in the tree is the root node. The nodes connected to it contain questions to be answered. Progress is made in line with the branches connecting the nodes and the answers given to the questions. The last node is the leaf node with results.

The decision tree method shows its superiority for inductive learning and in terms of predictive accuracy [16][17]. Estimation using decision trees has its advantages. Firstly, this approach is simple to operate and easy to explain to users. In addition, the problem of cost driver selection can be avoided by using the decision tree for feature subset selection in software effort estimation models.

2.2.2. Multi Criteria Decision Analysis (MCDA)

Multi-Criteria Decision Analysis is a structure used to resolve important and complex decision-making situations of decision makers [18]. MCDA is an "umbrella term to describe a collection of formal approaches which seek to take explicit account of multiple criteria in helping individuals or groups explore decisions that matter" [19].

Many MCDA methods have been proposed in literature. The most well-known of them are: AHP (Analytic Hierarchical Process) [20], TOPSIS (Ordering Simulation Technique in Ideal Solution) [21], PROMETHEE (The Preference Ranking Organization METHod for Enrichment of Evaluations) [22], and ELECTRE (Elimination and Choice Expressing the Reality) [23].

An MCDA decision making mechanism works with the following steps [24]: 1) Define the Decision Opportunity, 2) Identify Stakeholder Interests, 3) Build a Decision Framework, 4) Rate the Alternatives, 5) Weight Stakeholder Interests, 6) Score the Alternatives, 7) Discuss Results, Re-Score, Discuss Again, and Decide.

2.2.2.1. Fuzzy TOPSIS

Multi-criteria decision analysis (MCDA) methods are applied in many fields of study. Fuzzy TOPSIS, one of the many MCDA methods, has been found to be successfully applied by many researchers in many practical difficulties [25].

Hwang and Yoon proposed Technique for Order Performance by Similarity to Ideal Solution (TOPSIS) [26] and is the most used technique for solving MCDM problems. This method works on the following principle: The chosen alternative should have the shortest distance to the Positive Ideal Solution (PIS) and the longest distance to the Negative Ideal Solution (NIS). PIS is the solution that minimizes the cost criteria and maximizes the benefit criteria. NIS is the solution that maximizes the cost criteria and minimizes the benefit criteria. The preference order is made by using the closeness coefficient obtained from these distances.

Fuzzy TOPSIS applications are used in many areas. Some of these application areas are: cost, operation and maintenance cost, payback period (economic), land use (environmental), location problems etc [27]. With Fuzzy TOPSIS, decision makers can put their ideas into numerical form using a natural language and use this while evaluating alternatives [28].

2.3. ISBSG Dataset

In this work, ISBSG release 2016 R1.1 has been used [29]. According to the study [9], ISBSG dataset is widely used for software project estimations. The International Software Benchmarking Standards Group (ISBSG) maintains a data repository containing software project data from many organizations. The ISBSG aims to provide a wide range of project data from many sectors to organizations. These data can be used for awareness of trends, effort estimation, productivity benchmarking and comparing platforms and languages. The dataset contains over 7500 samples organized in many variables.

2.4. Chi-Squared Test

Pearson's chi-square test χ^2 is a statistical test that is applied to datasets to evaluate the probability that any observed difference between sets occurs by chance [30]. It is the most widely used of the many chi-square tests, which are statistical procedures where results are evaluated according to the chi-square distribution. It attempts to show that the distribution of results observed in a study is consistent with the theoretical distribution. To test this, a null hypothesis is put forward. While this test is performed using the p-value between two results, first Null and Alternate hypotheses are created. If the p-value is less than 0,05, the results are statistically significant at the 95% confidence interval. Therefore, the null hypothesis should be rejected. The Zero and Alternative Hypotheses for this study are presented below: Hypotheses Zero (H0): The two results are not different. Alternative Hypotheses (H1): The two results are different.

Calculating the value of the test-statistic is given in Figure 2.1 and the variables in the formula are explained below.

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

Figure 2.1. Chi Square Formula

χ^2 : Pearson's cumulative test statistic.

O_i : the number of observations of type i.

E_i : the expected (theoretical) frequency of type i, asserted by the null hypothesis that the fraction of type i.

n : is the number of results analyzed in an experiment.

1. The steps for calculating p value:
2. Determine the expected results of your experiment.
3. Identify the observed results of your experiment.
4. Determine the degrees of freedom of your experiment.
    a. Degrees of freedom = n – 1
5. Calculate Chi square results with formula in Figure 2.1.
6. Choose a significance level.
7. Use the chi-square distribution table (Table 2.1) to determine the p-value (with degrees of freedom and chi-square results). For example, if the Chi-square result for df=1 is 3, it means that the p value is in the range of 0.1 to 0.05.
8. Decide whether to reject or maintain the null hypothesis.

Table 2.1. Chi-square Distribution Table

					P					
df	**0.995**	**0.975**	**0.9**	**0.5**	**0.1**	**0.05**	**0.025**	**0.01**	**0.005**	**df**
1	.000	.000	0.016	0.455	2.706	3.841	5.024	6.635	7.879	1
2	0.010	0.051	0.211	1.386	4.605	5.991	7.378	9.210	10.597	2
3	0.072	0.216	0.584	2.366	6.251	7.815	9.348	11.345	12.838	3
4	0.207	0.484	1.064	3.357	7.779	9.488	11.143	13.277	14.860	4
5	0.412	0.831	1.610	4.351	9.236	11.070	12.832	15.086	16.750	5
6	0.676	1.237	2.204	5.348	10.645	12.592	14.449	16.812	18.548	6
7	0.989	1.690	2.833	6.346	12.017	14.067	16.013	18.475	20.278	7
8	1.344	2.180	3.490	7.344	13.362	15.507	17.535	20.090	21.955	8
9	1.735	2.700	4.168	8.343	14.684	16.919	19.023	21.666	23.589	9
10	2.156	3.247	4.865	9.342	15.987	18.307	20.483	23.209	25.188	10
11	2.603	3.816	5.578	10.341	17.275	19.675	21.920	24.725	26.757	11
12	3.074	4.404	6.304	11.340	18.549	21.026	23.337	26.217	28.300	12
13	3.565	5.009	7.042	12.340	19.812	22.362	24.736	27.688	29.819	13
14	4.075	5.629	7.790	13.339	21.064	23.685	26.119	29.141	31.319	14
15	4.601	6.262	8.547	14.339	22.307	24.996	27.488	30.578	32.801	15

3. EVALUATION APPROACH

The aim of this study was to provide estimators with a tool in selecting the best-fit software effort estimation method to enable more accurate effort estimation of software projects, which is an important step in software project planning. Accordingly, an evaluation approach based on Multi-Criteria Decision Analysis was created to select the most suitable software effort estimation method. While applying the MCDA, the core elements were determined as follows:

- *Problem*: Estimating software project effort accurately.
- *Requirements*: Developing software effort estimation model considering project requirements, data, and environmental dynamics.
- *Goal*: Selecting a SEE method that can best meet the requirements.
- *Criteria*: Various aspects required to develop a software effort estimation model in relation to project requirements.
- *Alternatives*: Software effort estimation methods that can meet the requirements in accordance with the determined criteria.
- *MCDA Tool*: An excel based decision matrix prepared using expert opinions.

3.1. Alternatives and Criteria

Alternatives. There are many different classifications of estimation methods in the literature [30]. In this study, we tried to select the most common effort estimation methods in classification and review studies. Although many review studies only examine the methods of one classification, we selected our alternatives by choosing methods from different classifications. While choosing our alternatives, we paid attention to be the most applied methods according to literature review studies. The methods we have chosen as an alternative in our study, with references to the motivating sources, are as follows:

- Neural Networks (NN) [6][8][11]
- Case-Base Reasoning (CBR) [6][8]
- Linear Regression (LR) [7][8][10]
- Analogy Based (AB) [7][11]
- Expert Judgement (EJ) [7][10][11]
- Support Vector Regression (SVR) [6][8]
- Decision Trees (DT) [6][8]

- Bayesian Networks (BN) [6][8]

Criteria. While preparing the questionnaire, the criteria that distinguish the SEE methods to evaluate were determined. These criteria play a role in determining how well the requirements match the methods. It is also aimed to determine the basic properties of the methods and to determine their compatibility with the project dynamics [13]. Criteria and related questions are shown in Table 3.1. The headings of criteria used in evaluation are explained below.

a) *Approach to construct method*: This criterion defines the method's approach to data dependency when configuring the SEE method. Methods estimate effort using historical data or estimation is done with different inputs independent of data.

b) *Data characteristics*: When creating the SEE method, the characteristics of data are decisive to choose the method to be successful. Addressing the limitations of the data will help in choosing the right method. The sub-criteria determined for data characteristics are as follows: type of input data, dataset size, and number of parameters.

c) *Data quality*: This criterion indicates the quality features of the data that will be used to construct the SEE method. These are uncertainty, missing values, and outliers. Uncertain data means that the data may be inaccurate, imprecise, untrusted or unknown. Besides, missing data for certain variables leads to poor estimations in some sensitive methods. Also, the outlier data can affect choosing the suitable method. An outlier is an observation that lies an abnormal distance from other values in a dataset.

d) *Method characteristics*: This criterion defines the characteristics of the methods to use to construct the SEE method. The method should be interpretable, easy to use (not complex), speedy, maintainable, and adaptive. Interpretability indicates that the user can understand the cause of any result. Ease of use (not being complex) is the degree of which the method is not complicated in design. Speed is the degree of which the method is built in a short time and performs fast in general. Maintainability is the degree of which the method is easy to manage in time. Being adaptive means that the method can accept new data without re-running the SEE method.

e) *Project context*: This criterion indicates the factors related to the context information of the project subject to SEE. The factors are software development life cycle, domain, size, and project type. Software development life cycle is an affecting factor to build the SEE method. Domain information is the expertise in the project area.

Project size information is considered as the size criterion. Project data type information represents cross-project or single-project options. A cross-project has multi single-projects. There are differences between these types in terms of project management and obtaining project information. Project data type has been added as a criterion for information that affects the method selection.

In addition, the experts who answered the expert-opinion survey were asked to add the criteria that they thought would affect the choice of the method and to add further methods that should be considered if any. They suggested that personnel parameters and project parameters should be added to the evaluation criteria. Fuzzy logic, soft computing methods, and sequential model optimization were suggested as the additional methods that should be considered in evaluation. Also, the experts advised that we should study with criteria and methods from the industry users' perspective, and not only the researchers' perspective.

Table 3.1. Criteria and Related Estimator Questionnaire

Criteria	Answer Type	Estimator Question
Approach to construct the model	Multiple	Do you want your model be dependent on data?
		Do you want to address human judgement?
Data characteristics	Multiple	Do you have categorical inputs?
		Do you have numerical inputs?
	Single	Is it the case that you have no data to train an SEE model?
		Do you have a small sized dataset to train an SEE model?
		Do you have a large sized dataset to train an SEE model?
	Single	Do you have past project data? If yes do you want to use it ?
Data quality	Single	Is there any uncertainty in the data? If yes, do you want to address the uncertainty?
	Single	Is there any missing values in the data? If yes, do you want to handle the missing values?
	Single	Is there any outlier in the data? If yes, do you want to handle these outliers?
Method characteristics	Single	Is it important that SEE model has high interpretability?
	Single	Is it important that SEE model has low complexity?
	Single	Is it important that SEE model can be build in a short time?
	Single	Is it important that SEE model has high maintainability?
	Single	Do you need that your model can accept new data without regenerating the model ?
Project context	Single	Do you have iterations in your software development life cycle? If yes, do you want to consider it in your SEE model?
	Single	Do you want to consider the domain information of your software project in your SEE model?
	Single	Do you want to consider the size information of your software project in your SEE model?
	Single	Do you have single-project data?
		Do you have cross-project data?

3.2. Expert-Opinion Survey and Estimator Questionnaire

A well-defined expert-opinion survey that collects the necessary data to specify the characteristics of the effort estimation methods was designed and conducted. The survey consisted of questions that allowed us to determine the weight of criteria defined above for the estimation methods. A group of experts having published studies on software effort estimation was selected and asked to participate in the survey. The experts have been doing academic studies for a long time in the field of effort estimation as seen in Figure 3.1. The expert-opinion survey resulted in answers by eight experts for three different question types; List selection (QT1), Ranking on Likert scale (QT2), Yes/No selection (QT3). The first type is list selection, for which possible answers are A, B, both A and B. The Likert scale used has the following answer options: very low, low, average, high, very high. The last one is Yes/No choice. As an example, the answers to these three question types for the Expert Judgment estimation method are given in Table 3.2.

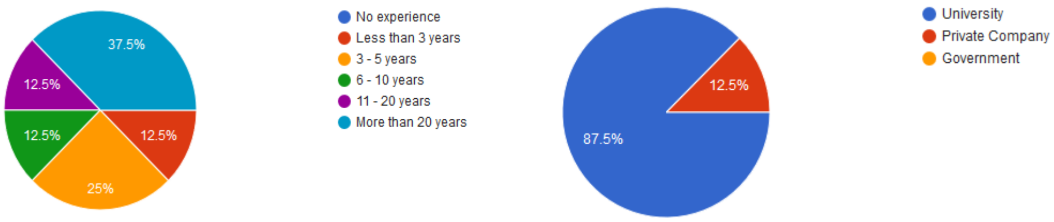


Figure 3.1. Year of Expertise in SEE and Organization Types of the Experts

Table 3.2. Answers to Three Types of Questions for Expert Judgment Estimation Method

Expert	**QT1:** Please select the convenient option on “Approach to Construct the SEE method” with the below methods.	**QT2:** To what extent do you think the following methods are “interpretable” by its users in SEE?	**QT3:** Do you think that iteration in software development life cycle is an affecting factor in SEE with the following methods?
E1	Based on human judgement	Low	Yes
E2	Based on human judgement	High	Yes
E3	Can address both	Very Low	Yes
E4	Based on human judgement	Average	Yes
E5	Based on human judgement	Very High	Yes
E6	Based on human judgement	*(not answered)*	*(not answered)*
E7	Based on human judgement	High	Yes
E8	Based on human judgement	High	No

4. MODELING THE DECISION MAKING STUDIES

4.1. Study-1: Decision Matrix

The decision matrix in Table 4.1 was created using the answers to the expert-opinion survey from eight experts. Estimator questions and weights in the decision matrix were derived from the expert-opinion survey results. We explain below the steps for generating and weighting three sample estimator questions (EQ) with respect to the three types of survey questions.

QT1. "Do you want your method be dependent on data?" (EQ1) and "Do you want to address human judgement?" (EQ2) questions were created of QT1 from the expert-opinion survey result. While determining the weight of EQ1 (W_{EQ1}), the number of "Dependent on data" and "Can address both" answers given was divided by the number of all answers to EQ1. Similarly, weight of EQ2 (W_{EQ2}) was determined by dividing the number of "Based on human judgment" and "Can address both" responses by the number of all responses to EQ2.

- W_{EQ1} = Count (Dependent on data) + Count (Can address both) / Count (All EQ1 answers)
  $W_{EQ1} = (0 + 1) / 8$
  $W_{EQ1} = 0.13$

- W_{EQ2} = (Count (Based on human judgement) + Count (Can address both)) / Count (All EQ2 answers)
  $W_{EQ2} = (7 + 1) / 8$
  $W_{EQ2} = 1$

QT2. The question "Is it important that SEE method has high interpretability?" (EQ12) was created of QT2 from the expert-opinion survey result. When determining the weight of EQ12, values in range [1-5] were assigned for the answers in range [Low-Very High]. Total weight of EQ12 ($W_{Total\text{-}EQ12}$) was calculated by summing the product of each answer (in [1-5]) by the weight value which was taken as the number of that answer given. Weighted sum of EQ12 (W_{EQ12}) was determined by dividing the total weight of EQ12 ($W_{Total\text{-}EQ12}$) by the sum of all EQ12 responses multiplied by the maximum weight value of 5.

- $W_{Total\text{-}EQ12}$ = 1 x Count (Very Low) + 2 x Count (Low) + 3 x Count (Average) + 4 x Count (High) + 5 x Count (Very High)
  $W_{Total\text{-}EQ12} = 1 \times 1 + 2 \times 1 + 3 \times 1 + 4 \times 3 + 5 \times 1$
  $W_{Total\text{-}EQ12} = 23$

- $W_{EQ12} = W_{Total\text{-}EQ12}$ / (Count (All EQ12 answers) x 5)
  W_{EQ12} = 23 / (7 x 5)
  W_{EQ12} = 0.66

QT3. The question "Do you prefer iteration in software development life cycle?" (EQ17) was created of QT3 from the expert-opinion survey result and its weight was determined by dividing the number of "Yes" answers by the number of all EQ13 answers.

- W_{EQ17} = Count (Yes) / Count (All QT3 answers)
  W_{EQ17} = 6 / 7
  W_{EQ17} = 0.86

The weights of the estimator questions were normalized to the range [0-1] to ensure that no criteria dominate other criteria during selection of an estimation method. In the calculation, the total number of answers given to the questions was used to eliminate the effect of the questions that were not answered by the experts. In this way, it is aimed to determine the estimation method selection not from the weight difference between the criteria, but from the weight difference between the key features of the methods.

Table 4.1. Decision Matrix

			Rating							
QID	**Estimator Question**	**Answer Type**	**EJ**	**AB**	**NN**	**BN**	**LR**	**DT**	**CBR**	**SVR**
EQ1	Do you want your model be dependent on data?	Multiple	0.13	0.75	1.00	1.00	1.00	1.00	1.00	1.00
EQ2	Do you want to address human judgement?		1.00	0.88	0.50	0.71	0.25	0.63	0.71	0.43
EQ3	Do you have categorical inputs?	Multiple	1.00	0.88	0.50	0.71	0.50	0.88	1.00	0.43
EQ4	Do you have numerical inputs?		0.71	0.75	0.88	0.71	1.00	0.88	0.88	0.86
EQ5	Is it the case that you have no data to train an SEE model?	Single	0.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ6	Do you have a small sized dataset to train an SEE model?		0.50	0.63	0.25	0.63	0.50	0.75	0.50	0.50
EQ7	Do you have a large sized dataset to train an SEE model?		0.25	0.38	0.75	0.38	0.50	0.25	0.50	0.50
EQ8	Do you have past project data? If yes do you want to use it ?	Single	0.67	0.83	0.83	0.60	0.67	0.67	0.67	0.60
EQ9	Is there any uncertainty in the data? If yes, do you want to address the uncertainty?	Single	0.57	0.57	0.63	0.73	0.57	0.60	0.57	0.68
EQ10	Is there any missing values in the data? If yes, do you want to handle the missing values?	Single	0.57	0.48	0.67	0.70	0.63	0.63	0.56	0.68
EQ11	Is there any outlier in the data? If yes, do you want to handle these outliers?	Single	0.76	0.68	0.63	0.70	0.57	0.63	0.60	0.72
EQ12	Is it important that SEE model has high interpretability?	Single	0.66	0.66	0.48	0.60	0.65	0.68	0.71	0.60
EQ13	Is it important that SEE model has low complexity?	Single	0.72	0.72	0.60	0.70	0.70	0.80	0.60	0.72
EQ14	Is it important that SEE model can be build in a short time?	Single	0.57	0.80	0.66	0.72	0.77	0.77	0.67	0.73
EQ15	Is it important that SEE model has high maintainability?	Single	0.60	0.57	0.71	0.72	0.69	0.57	0.66	0.77
EQ16	Do you need that your model can accept new data without regenerating the model ?	Single	0.67	0.83	0.57	0.50	0.57	0.57	0.33	0.71
EQ17	Do you have iterations in your software development life cycle? If yes, do you want to consider it in your SEE model?	Single	0.86	0.43	0.75	0.86	0.75	0.75	0.57	0.63
EQ18	Do you want to consider the domain information of your software project in your SEE model?	Single	0.83	0.83	0.57	0.67	0.57	0.71	0.83	0.57
EQ19	Do you want to consider the size information of your software project in your SEE model?	Single	0.23	0.20	0.21	0.20	0.17	0.18	0.19	0.20
EQ20	Do you have single-project data?	Single	1.00	1.00	0.80	0.75	0.80	1.00	0.75	0.80
EQ21	Do you have cross-project data?		0.75	0.50	1.00	1.00	0.60	0.60	1.00	1.00

An estimator questionnaire was derived from expert-opinion survey answers. The questionnaire is intended for use by a project staff who holds the role of an estimator and wants to carry out effort estimation in his/her project accurately. The expert-opinion survey was filled once by the experts and a decision matrix was prepared from it. Using having the decision matrix prepared, the estimator can use this matrix in order to select the most suitable estimation method for his/her need by answering a number of estimator questions (EQ).

In the decision matrix shown in Table 4.1, the first column (QID) refers to the identifier of the estimator question, the second column (Estimator Question) refers to the description of the estimator question, and the third column (Answer Type) refers to the way the question is answered. In that column 'Multiple' value is used for the criteria elicited by answering more than one question, and 'Single' value is used for the criteria elicited by answering only one question. In the other columns (Rating), the weights calculated from the expert-opinion survey as detailed above according to the estimator question types for the relevant estimation methods are given.

In the estimation process, an estimator answers the estimator questions by giving a value of 1 or 0, suitable for the question in each row. The answers are multiplied by the relevant method ratings, and the calculated scores for all questions are summed for each method to find the method scores. The method with a higher score is more suitable for estimation. Details of using the decision matrix in estimation process is explained in the next section.

4.2. Study-2: Two Step Decision Mechanism

4.2.1 Phase-1 Decision Tree Analysis

For the decision tree preparation, we first constructed a matrix given in Table 4.2. In this matrix, three criteria from two criteria groups ("Approach to construct", "Type of input data" and "Dataset size") were included. We chose these criteria in this matrix because the values that the relevant criteria could take were clearer, and therefore, the options were sharply clear. For this reason, the following matrix was created with the answers of the expert opinion questionnaire and a decision tree, which is given in Figure 4.1, was constructed based on this matrix.

A decision tree was prepared to be used in the first stage of the decision mechanism for the criteria determined according to the created decision matrix table (Table 4.2). Based on the prepared decision tree and the answers given by the estimator, a set of sub-options will be listed in line with their needs. Figure 4.1 shows the decision tree, which is the first stage of the decision mechanism. It is planned to choose between the methods at the end of the tree by moving forward. There are empty leaves on the tree. In this case, the selection will move to the previous level and continue until the alternative method is found. The study will proceed by choosing between the methods in the upper step.

Table 4.2. Matrix for Decision Tree

Method	**Approach to construct the model**	**Type of input data**	**Dataset size**
Expert Judgement	Human Judgement	Numerical, Categorical	Medium
Analogy Based	Can Address Both	Numerical, Categorical	Medium
Neural Networks	Dependent on Data	Numerical	Medium / Large
Bayesian Networks	Can Address Both	Numerical, Categorical	Medium
Linear Regression	Dependent on Data	Numerical	Medium / Large
Decision Trees	Can Address Both	Numerical, Categorical	Medium
Case-Base Reasoning	Can Address Both	Numerical, Categorical	Medium
Support Vector Regression	Dependent on Data	Numerical	Medium

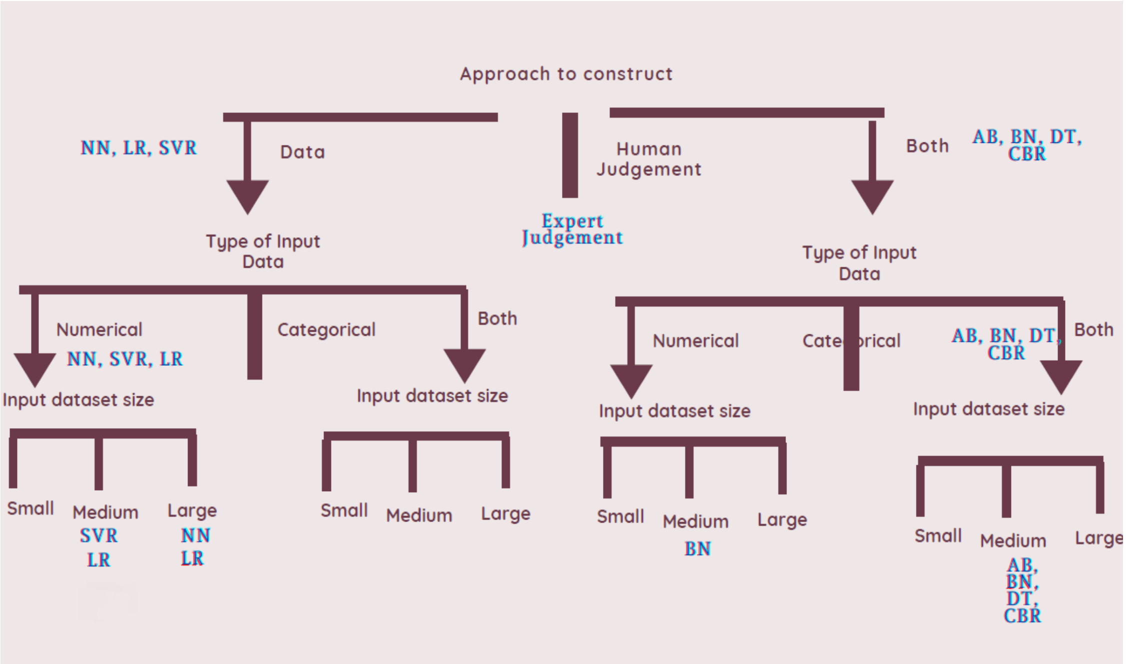


Figure 4.1. Decision Tree for Phase

4.2.2 Phase-2 Fuzzy TOPSIS

We used the Fuzzy TOPSIS calculation to answer the questions in the "Method Characteristics" and "Project Context" classification. Since the data types of the answers to the questions in these groups are interval, we preferred Fuzzy TOPSIS here.

According to expert opinion survey results, aggregated fuzzy importance weights were calculated for the questions in these groups. Since experts answered using linguistic variables in our survey, we had to convert them to fuzzy numbers. We used transform scales for this process. The linguistic variables used in our study and the fuzzy values we used for them are shown in the table (Table 4.3).

Table 4.3. Linguistic Variables and Fuzzy Values

Linguistic variables	Fuzzy values
Very Low	1, 1, 3
Low	1, 3, 5
Average	3, 5, 7
High	5, 7, 9
Very High	7, 9, 9

As in [13], we used triangular fuzzy numbers for five different linguistic variables in our questionnaire. We scaled these numbers from 1 to 9. We did the fuzzy calculation as follows [32]. J is the total decision maker (eight experts answered in our expert opinion survey.) W is the weight of a questions.

Step1. The decision matrix is normalized as follows:

$$r_{ij} = \frac{w_{ij}}{\sqrt{\sum_{j=1}^{J} w_{ij}^2}} \qquad j = 1, 2, 3, \ldots, J \quad i = 1, 2, 3, \ldots, n$$

Step 2. The weighted normalized matrix is obtained as follows:

$$v_{ij} = w_{ij} * r_{ij} \ , j = 1, 2, 3, \ldots, J \ , i = 1, 2, 3, \ldots, n$$

Step 3. The positive ideal solution and the negative ideal solution are found as follows:

$$A^* = \{v_1^*, v_2^*, \ldots, v_n^*, \}$$

$$A^- = \{v_1^-, v_2^-, \ldots, v_n^-, \}$$

Step 4. Calculate the distance of each alternative from positive and negative ideal solutions:

$$d_i^* = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^*)^2} \ , j = 1, 2, \ldots, J$$

$$d_i^- = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^-)^2} \ , i = 1, 2, \ldots, J$$

Step 5. The proximity coefficients of each alternative are calculated by the following formula:

$$CC_i = \frac{d_i^-}{d_i^* + d_i^-} \ , i = 1, 2, \ldots, J$$

Step 6. By comparing the CCi closeness coefficient values with each other, the ranking of the alternatives is obtained.

The base decision matrix used in the calculations is seen in Table 4.4. This matrix is the basic matrix created with the answers given to our last two question groups at expert opinion survey.

Table 4.4. Decision Matrix for Fuzzy TOPSIS Calculation

Combined Decision Matrix																					
Method	**Uncertainty**			**Missing Data**			**Outlier**			**Interpretability**			**Complexity**			**Accuracy**			**Maintainability**		
EJ	1.00	5.00	9.00	1.00	6.00	9.00	1.00	5.00	9.00	1.00	4.00	9.00	1.00	4.00	9.00	1.00	3.67	9.00	1.00	3.67	9.00
AB	1.00	5.00	9.00	1.00	5.00	9.00	1.00	3.00	7.00	1.00	4.50	9.00	1.00	5.00	9.00	3.00	7.67	9.00	1.00	3.67	7.00
NN	3.00	5.00	7.00	3.00	6.00	9.00	1.00	5.00	9.00	1.00	1.50	5.00	1.00	3.00	7.00	1.00	5.67	9.00	1.00	6.33	9.00
BN	3.00	6.33	9.00	3.00	5.00	7.00	3.00	5.00	7.00	1.00	4.00	9.00	1.00	5.00	9.00	1.00	6.33	9.00	3.00	6.33	9.00
LR	1.00	5.00	9.00	1.00	4.00	9.00	1.00	4.00	9.00	1.00	5.00	9.00	1.00	4.00	7.00	1.00	5.67	9.00	1.00	6.33	9.00
DT	1.00	4.00	7.00	1.00	4.00	7.00	1.00	3.00	5.00	1.00	3.50	7.00	3.00	5.00	7.00	3.00	6.33	9.00	3.00	6.00	9.00
CBR	1.00	5.00	9.00	1.00	4.00	7.00	1.00	5.00	9.00	1.00	6.00	9.00	1.00	4.00	7.00	1.00	5.00	9.00	1.00	5.00	9.00
SVR	3.00	5.00	7.00	1.00	4.00	7.00	3.00	5.00	7.00	1.00	3.50	7.00	3.00	5.00	7.00	3.00	6.33	9.00	3.00	7.00	9.00

Also, the rank table that emerges after the steps described above can be seen in Table 4.5. This rank table is a look up table that is calculated with best cases. The proposed MCDA aimed to rank the remaining option set after the decision tree calculation in line with the needs and requirements of the estimator.

Table 4.5. Ranking Among the SEE Methods

Method	CCi	Rank
EJ	0.629	5
AB	0.675	4
NN	0.625	6
BN	0.762	1
LR	0.737	2
DT	0.379	8
CBR	0.716	3
SVR	0.566	7

4.2.3. Software Effort Estimation Method Selection Tool

The Study-2 mentioned above was in the form of following the user over the decision tree and then finding the result by looking at the rank table. We developed a tool to improve and facilitate this decision. The Graphical User Interface (GUI) of the tool for SEE method selection is shown in Figure 4.2. The tool is a desktop application developed in Java language. With this application, the user gives the answers to the estimation questions and the tool proposes the suitable method by making progress in the tree according to the relevant answers and then choosing between the remaining methods through the calculated rank values. The estimator is expected to perform its operations without having knowledge of the decision tree and the associated rank table by using the tool from a simple user interface.

During the tool implementation, we have advanced our rank study, which consists of fuzzy calculations, and instead of the method that provides the best situation in method suggestion, method rankings created according to the requirements of the estimator are provided. Here, a dynamic rank table is created by making calculations according to the answers of the estimator. Considering the cases where the estimator answers "Yes", the fuzzy calculation is made and the rank table is calculated internally.

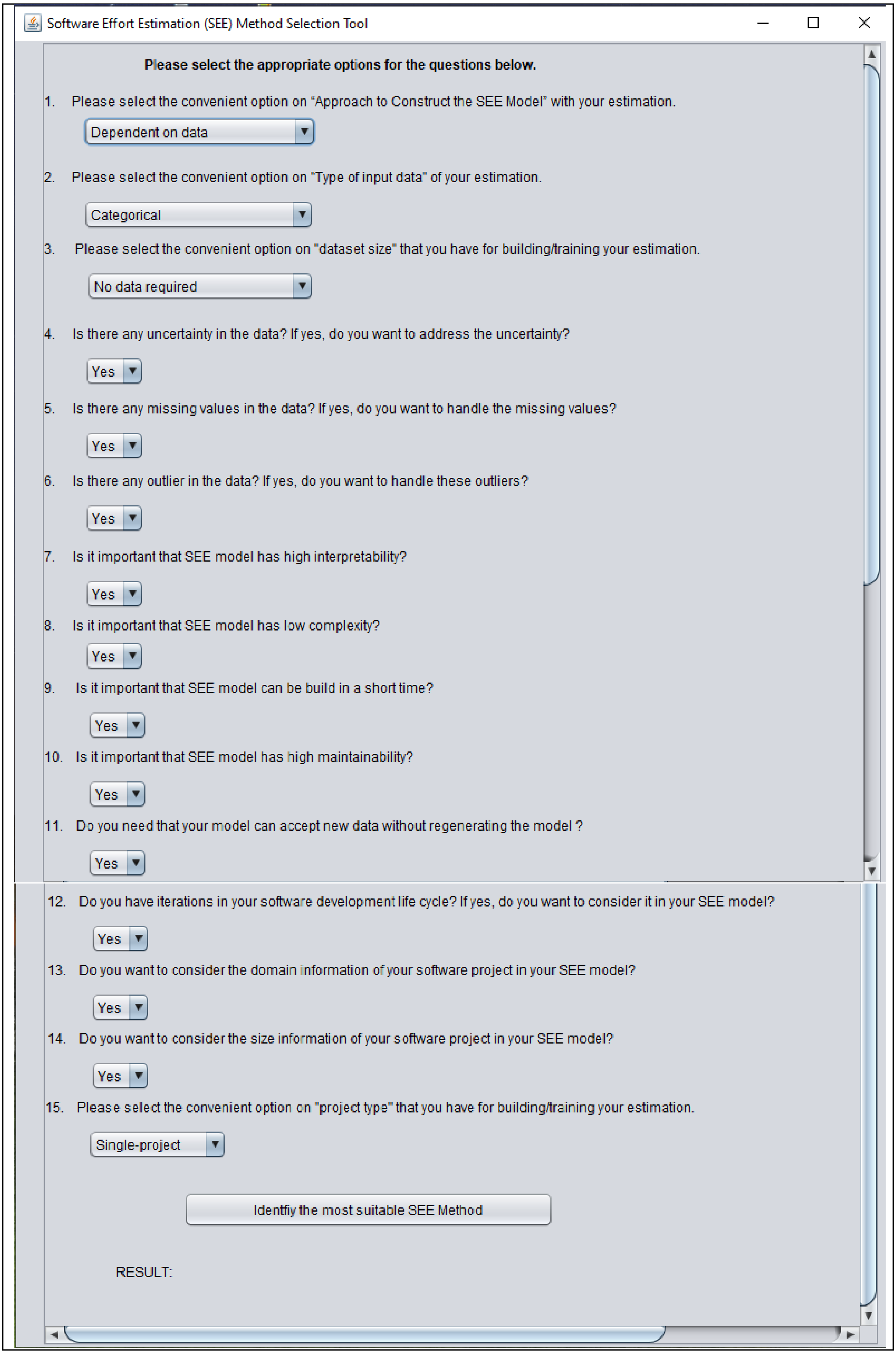


Figure 4.2. GUI of Software Effort Estimation Method Selection Tool

The operating steps of the proposed decision mechanism using the tool are summarized in Figure 4.3.

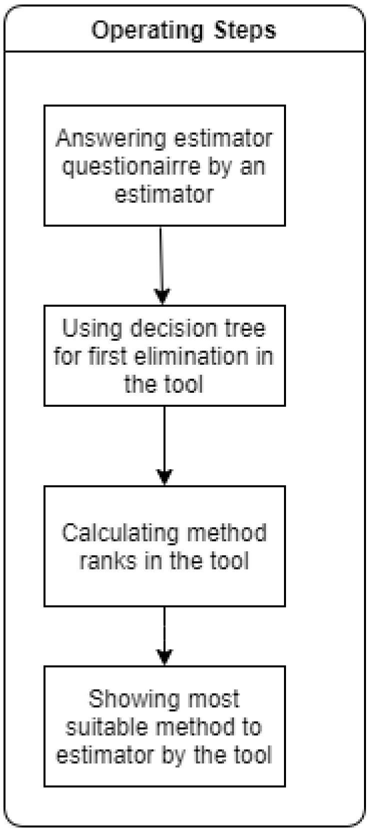


Figure 4.3. Operationg Steps of Decision Mechanism Using the Tool

5. EVALUATION

5.1. Evaluation of Study-1: Decision Matrix

The decision matrix described in Table 4.1 was detailed with an example evaluation in Table 5.1. The estimator questions were answered using the ISBSG dataset and a number of assumptions regarding the example estimation. In order to answer the questions, a project of a company was selected from the ISBSG dataset and its information was examined. While some of the answers were answered directly by using the dataset, some of them were answered by the researcher writing this thesis, according to the hypothetical estimation needs, considering the project information of the company. This information is shown in the “Reason” column in Table 5.1. The questions were answered as 1 for “Yes” and 0 for “No”, in accordance with the estimation context.

The reasons for each answer are as follows. The ISBSG dataset is used for the example estimation (EQ1) and the estimation model is preferred not to be dependent on human judgment (EQ2). That is why the answer to EQ1 is Yes, while that of EQ2 is No. Since there are categorical and numerical inputs in the ISBSG dataset, the answers given to EQ3 and EQ4 are Yes. The size of the dataset that can be used for training in the dataset is large, so the answer to EQ7 is Yes while the answers to EQ5 and EQ6 are No. The answer to EQ8 is Yes since other projects’ information can be used. The uncertainty information will not be addressed in the estimation, so the answer given to EQ9 is No. There are missing data in the dataset and this information will be handled in the estimation process (EQ10). There is an abnormal distance between the values in the dataset, so the preference is Yes for EQ11. The estimator does not need the estimation model to have high interpretability, low complexity, high maintainability and short built time. Therefore, the preferences are No for EQ12, EQ13, EQ14, and EQ15. The estimator does not need that the model can accept new data without regenerating so the answer for EQ16 is No. The iteration information from the dataset will not be handled in the estimation process (No for EQ17). The domain information will not be used in estimation, so the answer for EQ18 is No. The size information can be found in the dataset (Yes for EQ19). Finally, the estimator considers the project is a cross-project (No for EQ20 and Yes for EQ21).

After entering estimator responses, method scores were calculated using the rating values in the decision matrix. Summing all the scores in the relevant method column, the total score for each method was obtained in the last (SUM) row of the table. The answers and the total scores for each estimation method in our example evaluation can be seen in Table 5.1. The answers were given with respect to the characteristics of ISBSG dataset and the estimator's assumptions.

According to the decision matrix prepared with our approach, the most suitable effort estimation model with a score of 6.26 is the Neural Network (NN) method and then the Case Base Reasoning (CBR) method with a score of 6.20.

Table 5.1. Example Evaluation Using the Decision Matrix

				Score							
QID	Estimator Question	Reason	Preference (Yes=1, No=0)	EJ	AB	NN	BN	LR	DT	CBR	SVR
EQ1	Do you want your model be dependent on data?	Estimator	1	0.13	0.75	1.00	1.00	1.00	1.00	1.00	1.00
EQ2	Do you want to address human judgement?	Estimator	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ3	Do you have categorical inputs?	ISBSG	1	1.00	0.88	0.50	0.71	0.50	0.88	1.00	0.43
EQ4	Do you have numerical inputs?	ISBSG	1	0.71	0.75	0.88	0.71	1.00	0.88	0.88	0.86
EQ5	Is it the case that you have no data to train an SEE model?	ISBSG	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ6	Do you have a small sized dataset to train an SEE model?	ISBSG	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ7	Do you have a large sized dataset to train an SEE model?	ISBSG	1	0.25	0.38	0.75	0.38	0.50	0.25	0.50	0.50
EQ8	Do you have past project data? If yes do you want to use it ?	ISBSG	1	0.67	0.83	0.83	0.60	0.67	0.67	0.67	0.60
EQ9	Is there any uncertainty in the data? If yes, do you want to address the uncertainty?	Estimator	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ10	Is there any missing values in the data? If yes, do you want to handle the missing values?	ISBSG	1	0.57	0.48	0.67	0.70	0.63	0.63	0.56	0.68
EQ11	Is there any outlier in the data? If yes, do you want to handle these outliers?	ISBSG	1	0.76	0.68	0.63	0.70	0.57	0.63	0.60	0.72
EQ12	Is it important that SEE model has high interpretability?	Estimator	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ13	Is it important that SEE model has low complexity?	Estimator	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ14	Is it important that SEE model can be build in a short time?	Estimator	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ15	Is it important that SEE model has high maintainability?	Estimator	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ16	Do you need that your model can accept new data without regenerating the model ?	Estimator	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ17	Do you have iterations in your software development life cycle? If yes, do you want to consider it in your SEE model?	ISBSG	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ18	Do you want to consider the domain information of your software project in your SEE model?	Estimator	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ19	Do you want to consider the size information of your software project in your SEE model?	ISBSG	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ20	Do you have single-project data?	Estimator	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
EQ21	Do you have cross-project data?	Estimator	1	0.75	0.50	1.00	1.00	0.60	0.60	1.00	1.00
			SUM	4.83	5.24	6.26	5.80	5.47	5.53	6.20	5.79

Wen et al. [6] showed that NN and CBR methods with the usage of the ISBSG dataset are the most frequently used ones. They prepared a list for "distribution of the studies over the types of ML techniques". CBR and NN are at the top of the list. The research interest in CBR and NN methods have increased over the years compared to other methods. Also, these methods are more accurate than others when working with the ISBSG dataset. According to the mean magnitude of relative error (MMRE) values examined in the study, NN performed better than all other methods.

Marco et al. [8] systematically gathered the information of many studies that examined estimation methods in terms of accuracy. According to the results, the two best MMRE values of the studies performed with the NN method for the ISBSG dataset were calculated as 9.5 and 49. The two best MMRE values for CBR method with the same dataset were obtained as 53 and 52.32. It is seen from these results that NN achieves better estimation performance with ISBSG dataset. As in our study, NN is more suitable when a choice is made between NN and CBR.

Venkataiah et al. [33] examined which dataset and which methods were studied together in the literature. As a result of his analysis, he stated that one of the most worked methods with the ISBSG dataset is NN.

The above studies [6][8] review and list the MMRE values of the estimation methods from multiple studies. Although in some of these studies it is reported that methods other than NN and CBR give more accurate results (e.g. [34][35]), there are also studies that contain results that support the selection of these methods as suggested by our study (e.g. [36][37]). Therefore, we can say that the results obtained by our proposed approach in the sample evaluation is partially supported with the results and suggestions of studies in the latter group.

Nevertheless, comparing the selection of estimation methods in the studies based on the resulting MMRE values only might remain incomplete since the estimation process includes many requirements and assumptions other than the ones related to the dataset, as also considered in our evaluation approach. Accordingly, we need to create further estimation cases, or repeat the past estimation cases by applying our questionnaire when possible, to make more meaningful comparisons and to discuss the reliability of our evaluation approach.

5.2. Evaluation of Study-2: Two Step Decision Mechanism

5.2.1 Application of the Decision Mechanism

In this section, proposed second study was implemented for evaluation. For this purpose, firstly, the ISBSG dataset was examined and the columns related to effort estimation were determined. While deciding on the columns, it has been examined in the literature for the same purpose in the studies that selected the column in the ISBSG dataset [38][39][40].

The information in the columns, which is thought to provide correct estimation, was obtained by making various restrictions on the project data. These restrictions are as follows;

1. Consider only Data Quality A and B (column B in the dataset)
2. Consider only Year of Project: 2005-2015 (column D in the dataset)
3. Select not null values for normalized work effort attribute
4. Select the following attributes:
    - Functional Size,
    - Development Type
    - Development Platform
    - Language Type
    - Adjusted Function Points
    - Max Team Size
    - Organization Type
    - Primary Programming Language
    - Project Elapsed Time
    - Application Type
    - Normalized Work Effort
    - Summary Work Effort.

    Note: For estimation with methods, the selected attributes were: Functional Size, Adjusted Function Points, Project Elapsed Time, Summary Work Effort, and Normalized Work Effort.

When the above criteria were applied, 2221 project records were obtained. After preparing the input dataset, based on the selected data, the prepared estimator questions (given in Table 3.1) were answered. The ISBSG dataset was used for the example estimation, so EQ1 was answered as "Yes" for the SEE model to depend on data. With this first question, the sub-tree on the left was reached through the decision tree. The question of "type of input data" was given a numerical answer (with EQ4), and then the answer to the input dataset question was "Medium" (with EQ6). Because of there were 2221 records in the dataset we obtained we choose “Medium”. The ISBSG dataset contains 7518 records in the unfiltered state. The Desharnais dataset contains 81 records, while the NASA dataset contains 93 records [41]. Our input dataset that we prepared, according to these datasets, was considered as of medium size. The option set obtained as a result of the decision tree became "SVR" and "LR" methods. When the Fuzzy TOPSIS study between these two alternative models was chosen as described above, we saw that the LR model was found to be more successful than the SVR model.

5.2.2. Experimental Study

Linear Regression (LR) and Support Vector Regression (SVR) methods were constructed and applied on ISBSG dataset by using WEKA tool [42]. Also, NN method was constructed and applied on Java Neural Network Framework [43]. In method studies, a 10-fold cross-validation approach was adopted in the training and testing stages. The results of these studies are shown in the Table 5.2 [44] below.

Table 5.2. Method Implementation Results

Method	**Mean absolute error**	**Relative absolute error**
LR	215.8	6.53%
SVR	216.5	6.68%
NN	364	11.3%

After the study with the dataset created in section 5.2.1, the obtained results were as in the table. According to the results, the most effective method was LR. Then SVR, and NN as the least recommended method. In our study, as we progressed through our tree, we had LR and SVR in the tree branches, and as a result of the second step of the study, LR was suggested among the two models. Accordingly, the results of the experimental study appear to be compatible with the results of our evaluation approach. In addition, the NN model, which was eliminated by staying in the other branch in the last step while moving on the tree, was also run with the relevant dataset and was less successful compared to the other two models. This ultimately supports progress in the tree and that we have reached the right node.

5.3. Extended Evaluation of Study-2 by Multiple-Case Study

We conducted a multiple-case study to examine and support the accuracy of the Study-2. Our aim with this case study was to investigate the accuracy of our proposed study on four different datasets created from within the ISBSG dataset [29]. We applied the embedded multi-case design as suggested by Yin [45]. The design of the four cases and their embedded units of analysis are shown in Figure 5.1. With the four datasets we created for the different cases, we aimed to investigate whether the results suggested by using our approach and the experimental results obtained from the relevantly created datasets were conformant.

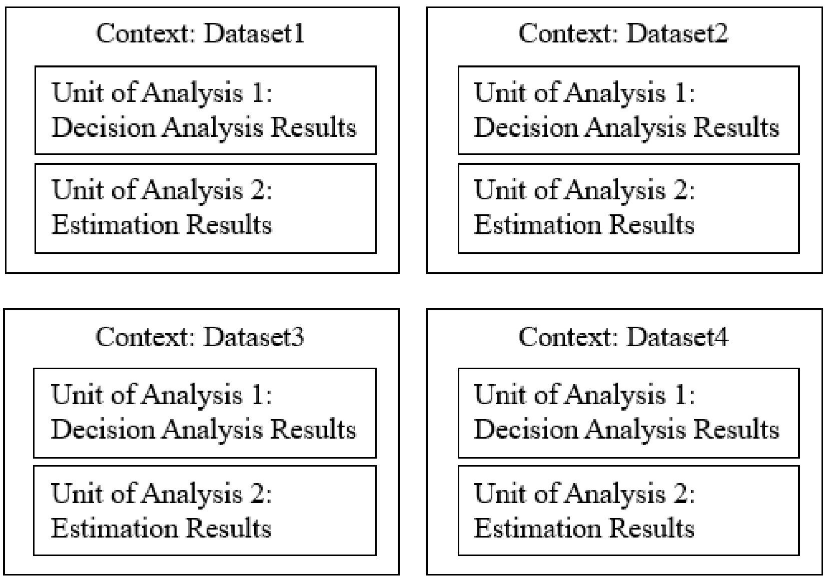


Figure 5.1. Embedded Multi-Case Study Design

In addition to the validation of the proposed approach and its results, we wanted to evaluate the execution of our Study-2 and the use of the tool that we developed by an outer estimator, who have worked in the field of effort estimation, in the three cases to be carried out. As an overview in Table 5.3, we provide a list of the cases, estimators (researcher or independent estimator) and implementation methods (manually or by tool).

Further details of the case study are explained in the following sections. In Section 5.3.1, the context of the study and the prepared datasets are explained. In Section 5.3.2, research questions of the study are given. In Section 5.3.3, answers to the research questions are elaborated and also, decision analysis processes and results are provided.

Table 5.3. List of the Cases, Estimators and Implementation Methods

Case	**Estimators and Implementation methods**
CS1	1.1) Researcher manually 1.2) Independent estimator manually 1.3) Independent estimator by tool
CS2	2.1) Independent estimator by tool
CS3	3.1) Researcher by tool
CS4	4.1) Researcher by tool

5.3.1. Context

Comparisons of the rank values formed as a result of the analysis study we prepared in our Study-2 with the MMRE values from the applications of the related methods were aimed in a multiple-case study context including four different datasets regarding the four cases. All prepared datasets were the ones obtained by filtering from the ISBSG database.

Dataset 1 used in Case Study 1 (CS1) was the dataset used in the first approach of Study-2 (explained in section 5.2.1). As mentioned before, it was obtained by filtering on many fields. Other datasets were prepared from within Dataset 1 as described below:

- Dataset 2, used in Case Study 2 (CS2), was prepared by selecting recent data of 1827 rows with respect to the year field covering the period of 2010-2015. With this data set, it was aimed to work with more recent data. It was thought that it would be healthier to rely on the accuracy of the data in line with the recency of the data due to technological advancements in project-supporting infrastructures.

- Dataset 3, used in Case Study 3 (CS3), was a 1000-row dataset prepared by including all the columns containing categorical data. With this dataset, it was aimed to use the characteristics of categorical data in the estimator's answers by adding the previously eliminated columns.
- Dataset 4, used in Case Study 4 (CS4), was a 160-row dataset obtained over Dataset 3 by selecting the rows having values of "Enhancement" in "Development Type" field and "Telecommunications" in "Organization Type" field. This dataset was aimed to work with due to its small size.

5.3.2. Research Questions

The purpose of the multiple-case study was to investigate the validity of our proposal by answering the following Research Questions (RQs):

- RQ1: Does the number of experts involved in creating the knowledge base and the decision matrix have an effect on the results obtained?
- RQ2: Does the approach provide effective results?
- RQ3: Does the tool validly support the decision analysis approach?
- RQ4: Is the approach/tool usable by an independent estimator?

We aimed to answer the above questions by comparing the results of the decision analysis we carried out using the tool in the context we explained, and the application results of the relevant method candidates. Our motivation for the related cases was that the results we got in the calculations were conformant due to the characteristics of the datasets we used.

In the next section (and its sub-sections), we elaborate on the answers to the RQs. In Table 5.4, we provide traceability of the RQs to the cases that were carried out to answer them.

Table 5.4. Traceability of RQs to the Cases

RQ#	Case#
RQ1	CS1.1, CS1.2, CS1.3
RQ2	CS1.1, CS1.2, CS1.3, CS2, CS3, CS4

RQ3	CS1.2, CS1.3
RQ4	CS1.2, CS1.3, CS2

5.3.3. Answers to the Research Questions

5.3.3.1. The effect of number of experts on decision matrix results (RQ1)

The expert opinion questionnaire used in Study-1 and Study-2 regarding CS1.1 was created by using the answers from 8 experts. Later, the number of experts from whom we obtained the estimation method information with the questionnaire (in Appendix-1) increased to 11. The second set of ranking values obtained by the decision matrix with the answers of 11 experts used in CS1.2 and CS1.3. Thus, the number of experts was increased by 37.5%. It was observed that the two sets of ranking values of the estimation methods by using two different decision matrices (with 8 and 11 experts) were very close to each other, as given in Table 5.5. The first set of ranking values obtained by using the decision matrix with the answers of 8 experts was very similar to the second set of ranking values obtained by the decision matrix with the answers of 11 experts, except the slight changes (5 or 6) in the ranking values of two methods, namely EJ and NN. Thus, we conclude that feedback received from different number of experts was consistent. It also showed the reliability of the expert opinion survey and its answers on which we based our proposal.

Table 5.5. Method Ranking using Decision Matrices obtained by 8 and 11 Experts for CS1

Method	CCi with 8 expert answers	Rank with 8 experts	CCi with 11 expert answers	Rank with 11 experts
EJ	0.629	5	0.559	6
AB	0.675	4	0.591	4
NN	0.625	6	0.585	5
BN	0.762	1	0.716	1
LR	0.737	2	0.639	2

DT	0.379	8	0.299	8
CBR	0.716	3	0.603	3
SVR	0.566	7	0.471	7

5.3.3.2. Effectiveness of the results suggested by the approach (RQ2)

With this research question, we took the evaluation of our Study-2 one step further and aimed to extend it with CS2, CS3, and CS4. The selections by following the proposed approach should be in line with the actual prediction results obtained. CS2 was carried out with tool by an independent estimator. CS3 and CS4 was carried out with tool by researcher. CS3 and CS4 are explained in this section. CS1.1, CS1.3 and CS2 will be explained in Section 5.3.3.4. Experimental results of these studies are also explained [44].

CS3 Results:

For CS3, within Unit of Analysis 1, the answers given by the estimator to the questions are given in Table 5.6. Then, with these answers, the rank values of the suggested methods and alternatives obtained by the tool are also displayed in Table 5.7.

Table 5.6. Estimator Answers for CS3

QID	Estimator Question	Preference
EQ1	Do you want your model be dependent on data?	1
EQ2	Do you want to address human judgement?	0
EQ3	Do you have categorical inputs?	1
EQ4	Do you have numerical inputs?	1
EQ5	Is it the case that you have no data to train an SEE model?	0
EQ6	Do you have a small sized dataset to train an SEE model?	0
EQ6	Do you have a medium sized dataset to train an SEE model?	1
EQ7	Do you have a large sized dataset to train an SEE model?	0
EQ8	Do you have past project data? If yes do you want to use it?	1
EQ9	Is there any uncertainty in the data? If yes, do you want to address the uncertainty?	1
EQ10	Is there any missing values in the data? If yes, do you want to handle the missing values?	1
EQ11	Is there any outlier in the data? If yes, do you want to handle these outliers?	1

EQ12	Is it important that SEE model has high interpretability?	1
EQ13	Is it important that SEE model has low complexity?	1
EQ14	Is it important that SEE model can be build in a short time?	1
EQ15	Is it important that SEE model has high maintainability?	1
EQ16	Do you need that your model can accept new data without regenerating the model?	1
EQ17	Do you have iterations in your software development life cycle? If yes, do you want to consider it in your SEE model?	1
EQ18	Do you want to consider domain information of your project in your SEE model?	1
EQ19	Do you want to consider size information of your project in your SEE model?	1
EQ20	Do you have single-project data?	0
EQ21	Do you have cross-project data?	1

Table 5.7. Ranking of Methods for CS3

Method	CCi	Rank
EJ	0.448	7
AB	0.456	6
NN	0.498	4
BN	0.805	1
LR	0.511	3
DT	0.334	8
CBR	0.480	5
SVR	0.519	2

Among the remaining methods after the first elimination stage at the tree, the top ranked methods according to these values were LR, SVR and NN. As a continuation of CS3 and within Unit of Analysis 2, the MMRE values obtained as a result of running the suggested estimation methods on Dataset 3 by Weka tool are shown in Table 5.8. Since our inputs in Dataset 3 contained categorical values, our tool was unable to reach the leaf nodes of the left branch of the decision tree. Rather, it stayed at the top and made suggestions by comparing all the methods on the left branch. Accordingly, our tool suggested the "Support Vector Machine" as the most suitable method for CS3. As seen from the results given in Table 5.8, mean absolute error and relative absolute error values of the estimations run were consistent with what the tool suggested, with the least values belonging to SVR. The ordering among the alternative methods was also congruent.

Table 5.8. Method Implementation Results for CS3

Method	Mean absolute error	Relative absolute error
LR	409.5	12.05%
SVR	240.8	7.09%
NN	425.1	12.51%

CS4 Results:

For CS4, within Unit of Analysis 1, the answers given by the estimator to the questions are given in Table 5.9. Then, with these answers, the rank values of the suggested methods and alternatives obtained by the tool are also displayed in Table 5.10.

Table 5.9. Estimator Answers for CS4

QID	Estimator Question	Preference
EQ1	Do you want your model be dependent on data?	1
EQ2	Do you want to address human judgement?	0
EQ3	Do you have categorical inputs?	1
EQ4	Do you have numerical inputs?	1
EQ5	Is it the case that you have no data to train an SEE model?	0
EQ6	Do you have a small sized dataset to train an SEE model?	1
EQ6	Do you have a medium sized dataset to train an SEE model?	0
EQ7	Do you have a large sized dataset to train an SEE model?	0
EQ8	Do you have past project data? If yes do you want to use it ?	0
EQ9	Is there any uncertainty in the data? If yes, do you want to address the uncertainty?	1
EQ10	Is there any missing values in the data? If yes, do you want to handle the missing values?	1
EQ11	Is there any outlier in the data? If yes, do you want to handle these outliers?	1
EQ12	Is it important that SEE model has high interpretability?	1
EQ13	Is it important that SEE model has low complexity?	1
EQ14	Is it important that SEE model can be build in a short time?	1
EQ15	Is it important that SEE model has high maintainability?	1

EQ16	Do you need that your model can accept new data without regenerating the model ?	1
EQ17	Do you have iterations in your software development life cycle? If yes, do you want to consider it in your SEE model?	1
EQ18	Do you want to consider domain information of your project in your SEE model?	1
EQ19	Do you want to consider size information of your project in your SEE model?	1
EQ20	Do you have single-project data?	0
EQ21	Do you have cross-project data?	1

Table 5.10. Ranking of Methods for CS4

Method	CCi	Rank
EJ	0.448	7
AB	0.456	6
NN	0.498	4
BN	0.805	1
LR	0.511	3
DT	0.334	8
CBR	0.480	5
SVR	0.519	2

As a continuation of CS4 and within Unit of Analysis 2, the MMRE values obtained as a result of running the suggested estimation methods on Dataset 4 by Weka tool are shown in Table 5.11. This case was studied with a small data set. Our tool was unable to reach to the leaf nodes of the left branch of the decision tree, first it switched to the left arm with the answer of "Data" for EQ1, and then it moved to the left arm with the answer of "Numerical" for EQ4. As a result of this progress, the tool made suggestions by sorting between LR, NN, SVR. The tool suggested the "Support Vector Machine" as the most suitable method for CS4. As seen from the results given in Table 5.11, mean absolute error and relative absolute error values of the estimations run were consistent with what the tool suggested, with the least values again belonging to SVR. The ordering among the alternative methods was also congruent.

Table 5.11. Method Implementation Results for CS4

Method	Mean absolute error	Relative absolute error
LR	409.52	12.05%
SVR	240.81	7.09%
NN	414.8	12.73%

5.3.3.3. Validity of the SEE Method Selection Tool (RQ3)

In answering RQ3, our expectation was that the decision obtained manually should be the same with the one obtained by the tool. In order to test the validity of the tool we developed, CS1.2 and CS1.3 were studied as we initially carried out Study-2. The independent estimator manually operated CS1.2 and as a result, "Linear Regression" was recommended as the most suitable method. When he worked over the tool with the same estimator answers (in CS1.3), the tool also offered "Linear Regression" as the best suggestion, as shown in Figure 5.2. This indicated that the suggestion made by the tool developed was identical to the one done manually and that the operation of the tool was valid.

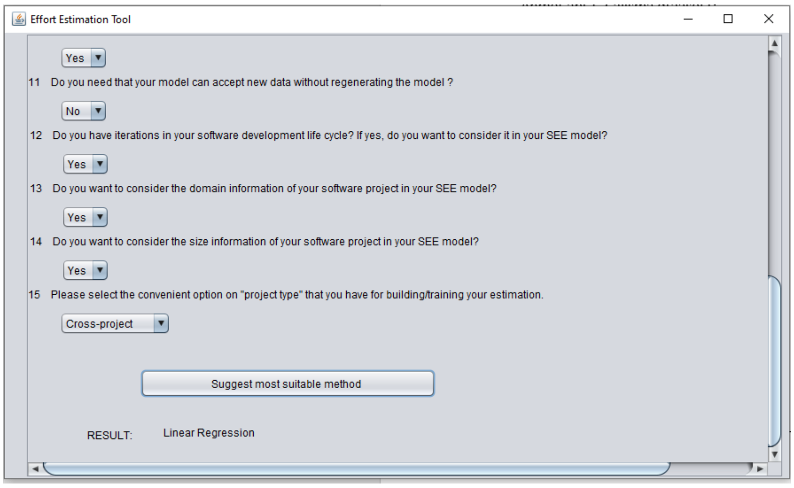


Figure 5.2. SEE Method Suggestion by using Tool for CS1.3

5.3.3.4. Usability of the approach/tool by an independent estimator (RQ4)

With RQ4, we wanted to investigate the usability of our approach/tool by an independent estimator. Here, our expectation was that the approach/tool should be usable by estimators other than the researcher. We sent our Dataset 1 to an estimator who has knowledge in this field and asked him to carry out a manual decision analysis by following our approach (in CS1.2). The estimator also performed the study over the tool (in CS1.3). Then, we asked him to perform estimation for CS2, again over the tool. During these processes, the independent estimator was provided with estimation data and information about the context of the multiple-case study, but no guidance was given on the answers for the cases. The answers and results obtained from the cases CS1.2, CS1.3 and CS2 are explained in the following paragraphs.

CS1 Results:

The answers that the independent estimator provided for CS1.2 and CS1.3 are shown in Table 5.12. When he operated decision analysis manually with these answers (in CS1.2), he reached the "SVR" and "LR" methods by advancing on the tree, and then reached the "LR" result with the rank table provided in Table 4.5. The experimental results of the related study supported this result as explained earlier in Section 5.2. In addition, when he operated decision analysis with the same answers over the tool (in CS1.3), he was suggested to use the "LR" method as the most suitable alternative. Therefore, the independent estimator was able to use the proposed approach both manually and over the tool, and got the same results as the researcher had in Study-2 (or in CS1.1).

Table 5.12. Answers of Independent Estimator for CS1

QID	Estimator Question	Preference
EQ1	Do you want your model be dependent on data?	1
EQ2	Do you want to address human judgement?	0
EQ3	Do you have categorical inputs?	0
EQ4	Do you have numerical inputs?	1
EQ5	Is it the case that you have no data to train an SEE model?	0
EQ6	Do you have a small sized dataset to train an SEE model?	0
EQ7	Do you have a large sized dataset to train an SEE model?	1

EQ8	Do you have past project data? If yes do you want to use it ?	1
EQ9	Is there any uncertainty in the data? If yes, do you want to address the uncertainty?	1
EQ10	Is there any missing values in the data? If yes, do you want to handle the missing values?	1
EQ11	Is there any outlier in the data? If yes, do you want to handle these outliers?	1
EQ12	Is it important that SEE model has high interpretability?	1
EQ13	Is it important that SEE model has low complexity?	0
EQ14	Is it important that SEE model can be build in a short time?	0
EQ15	Is it important that SEE model has high maintainability?	1
EQ16	Do you need that your model can accept new data without regenerating the model ?	0
EQ17	Do you have iterations in your software development life cycle? If yes, do you want to consider it in your SEE model?	1
EQ18	Do you want to consider domain information of your project in your SEE model?	1
EQ19	Do you want to consider size information of your project in your SEE model?	1
EQ20	Do you have single-project data?	0
EQ21	Do you have cross-project data?	1

CS2 Results:

For CS2, within Unit of Analysis 1, the answers given by the estimator to the questions are given in the Table 5.13. Then, with these answers, the rank values of the suggested methods and alternatives obtained by the tool are also displayed in Table 5.14. As a continuation of CS2 and within Unit of Analysis 2, the MMRE values obtained as a result of running the suggested estimation methods on Dataset 2 by Weka tool are shown in Table 5.15.

Table 5.13. Answers of Independent Estimator for CS2

QID	Estimator Question	Preference
EQ1	Do you want your model be dependent on data?	1
EQ2	Do you want to address human judgement?	0
EQ3	Do you have categorical inputs?	0
EQ4	Do you have numerical inputs?	1
EQ5	Is it the case that you have no data to train an SEE model?	0
EQ6	Do you have a small sized dataset to train an SEE model?	0
EQ6	Do you have a medium sized dataset to train an SEE model?	1
EQ7	Do you have a large sized dataset to train an SEE model?	0
EQ8	Do you have past project data? If yes do you want to use it ?	1

EQ9	Is there any uncertainty in the data? If yes, do you want to address the uncertainty?	1
EQ10	Is there any missing values in the data? If yes, do you want to handle the missing values?	1
EQ11	Is there any outlier in the data? If yes, do you want to handle these outliers?	1
EQ12	Is it important that SEE model has high interpretability?	1
EQ13	Is it important that SEE model has low complexity?	0
EQ14	Is it important that SEE model can be build in a short time?	0
EQ15	Is it important that SEE model has high maintainability?	1
EQ16	Do you need that your model can accept new data without regenerating the model ?	0
EQ17	Do you have iterations in your software development life cycle? If yes, do you want to consider it in your SEE model?	1
EQ18	Do you want to consider domain information of your project in your SEE model?	1
EQ19	Do you want to consider size information of your project in your SEE model?	1
EQ20	Do you have single-project data?	0
EQ21	Do you have cross-project data?	1

Table 5.14. Ranking of Methods for CS2

Method	CCi	Rank
EJ	0.330	6
AB	0.213	8
NN	0.381	4
BN	0.842	1
LR	0.805	2
DT	0.349	5
CBR	0.309	7
SVR	0.727	3

Table 5.15. Method Implementation Results for CS2

Method	Mean absolute error	Relative absolute error
LR	209.24	7.3%
SVR	201.75	7.04%
NN	316.8	11.07%

With this case, the tool was able to reach to the leaf nodes of the left branch of the decision tree. First, it switched to the left arm with the answer of "Data" for EQ1, and then it moved to the left arm with the answer of "Numerical" for EQ4. As a result of this progress, the tool made suggestions by sorting between LR and SVR.

At the end of the analysis, our tool suggested the "Linear Regression" as the most suitable method for CS2. There was a difference of 0.26% between “LR” and “SVR” in the relative absolute error values obtained, and the “SVR” method seemed more effective based on these values. However, when the rank values and MMRE values were checked, it was seen that the scores of the two methods for the Dataset 2 were very close and it was decided that the results could be considered compatible with the calculations below.

A chi-square test (explained in Section 2.4) was applied to show that the difference between the Mean Absolute Error values of the LR and SVR methods was statistically insignificant. Observed and expected values required for the test were determined first (Table 5.16). The method implementation results of LR and SVR were used as the Observed value. The result of the LR method implementation (209.24) was used as the expected value for both methods. In order to show that the difference was insignificant in this way, the expected values of both methods were determined the same and their compatibility with the observed values was tested. The fact that the p value calculated with these parameters was greater than 0.05 also indicated that the expected values complied with the observed values, that is, the difference was insignificant.

Table 5.16. Observed and Expected Values of LR and SVR Methods

	LR	SVR
Observed values	209.24	201.75
Expected values	209.24	209.24

Since there were two parameters in our test, LR vs SVR, the value of n was 2 and hence, degrees of freedom 1. Chi square calculation for this test is shown in the calculation (5.1). The chi-square value was found to be 0.268. By following the degrees of freedom row from left to right in Table 2.1, the p-value range for our chi-square result was determined as 0.01 - 0.05. According to the values in test calculations given in Table 5.17, since the detected p value was greater than 0.05, the expected values were appropriate, that is, we could say that the difference between the method results was statistically insignificant.

$$\frac{(201.75-209{,}24)^2}{209{,}24}+\frac{(209.24-209{,}24)^2}{209{,}24}=0{,}268 \tag{5.1}$$

Table 5.17. Values in Test Calculations

Degrees of freedom	2 – 1 = 1
Chi-square	0.268
Significance Level	0,05
P value	3.841

In addition, the interval (X) where the SVR method was statistically insignificant is calculated as follows (5.2). For the value to be insignificant, the p-value must be greater than 0.05. Accordingly, we could say that in cases where the SVR result was smaller than the test result (3.841), which gives p value of 0.05 for df value of 1 in Table 2.1, the difference was insignificant.

Consequently, our SVR result is within the X range where the difference between the SVR result and the LR result is considered negligible (5.3). Since the SVR result of 201.75 was in this range, we could say that the difference was insignificant.

$$3{,}841 > \frac{(X-209{,}24)^2}{209{,}24} \tag{5.2}$$

$$\begin{gathered}28{,}35 > X-209{,}24 \quad \&\& \quad 209{,}24-X > 28{,}35\\ Result\colon\ 180{,}89 < X < 237{,}59\end{gathered} \tag{5.3}$$

5.3.3.5. Overall results of the multiple-case study

As a summary, it was observed from the results of multiple-case study that the proposed decision analysis mechanism worked correctly and consistently. The answers of the experts participating in the study were consistent and reliable (in RQ1), so that the result obtained did not change when new experts were added. For the accuracy of the approach proposed using this information (in RQ2), it was seen that in three of the four cases (CS1.3, CS3 and CS4), the methods suggested by the approach were identical with the ones best performing in actual estimations and in only CS2, a different result was obtained. However, since the estimation results changed the ranking with only 0.26% difference in relative absolute error, it was considered negligible. In addition, it was observed (in RQ3) that the tool we developed to support the execution of the approach gave results compatible with the manual execution in our previous Study-2 and also consistent results in our new cases. Thanks to the tool, method selection could be made with a simple user interface. Finally, to investigate the usability of our approach/tool by an external estimator (in RQ4), we asked an expert who has worked in this field to use our approach/tool. Having the same results by the independent estimator related to this study increased our confidence in the study.

In addition, the independent estimator who carried out CS1.3 and CS2 assessed his experience in using the approach and the tool with a questionnaire given in Appendix 2. The criteria to assess the approach in the questionnaire included comprehensibility, ease of use, simplicity, clearness, internal consistency, need of SEE knowledge, and need of context information of the cases. Also, the criteria to assess the tool in the questionnaire included understandability, ease of use, simplicity, fairness of GUI design, and operability. As a result of the assessment, while the approach was appreciated in terms of simplicity and internal consistency, it was seen as open to improvement in terms of comprehensibility, ease of use, and clearness. Moreover, the tool was found as open to improvement in terms of fairness of GUI design and operability in the results, while it was found to be sufficient in terms of understandability and ease of use. However, these evaluations will be healthier when the tool will be used by multiple estimators.

5.3.4. Threats to Validity

Threats that would affect the validity of the case study were systematically identified. We used the work of Wohlin et al. [46] to identify these threats. The four main types of threads that may threaten the validity of the case study are discussed in the following paragraphs.

Internal validity:

Internal validity is concerned with how we are sure of the factors that influence a cause-effect relationship established in a study. The validity of our study may be affected by the lack of objectivity of the researcher. In order to prevent the researcher's bias from becoming a threat, support was received from an expert in the field of effort estimation. We tried to increase internal validity by adding his point of view to our study. In addition, the validation of our study was done on the datasets we determined. While the data set is being prepared, situations such as not being able to get uniformly distributed data and inconsistency of the data may pose a threat. In order to avoid this, a decision analysis study was carried out by preparing four different datasets. In these datasets, it was aimed to reduce the threat by using different criteria and choosing data with different characteristics. Still, deriving the four datasets from only the ISBSG dataset can be considered as a threat. In addition, the fact that the estimations made by using these datasets could reach the nodes only on one side of the decision tree might also have posed a threat. It is suggested in future studies that the approach should be tested with estimation data that can reach the other side of the decision tree.

Construct validity:

Construct validity focuses on the relationship between the actual observations of the study and the structure of the study. The result we get may not be the result we thought we measured. To increase construct validity in this study, we completed the first stage by filtering the literature studies in the selection of the methods and criteria we used. Then, we received support from experts who have been working in software effort estimation for a long time, by conducting an expert survey to ensure that we have access to reliable information. Moreover, the results we obtained did not change despite the increase in the number of experts participating in our study, suggesting that construct validity was provided. By supporting our selection approach by a tool, misdirection in the decision was prevented. Thus, the validity of the study was ensured in terms of consistency. However, it may not be completely objective when responding to the estimator questionnaire using our selection approach since it assumes that the estimator provides valid answers with respect to the dataset that s/he works with. This introduces some subjectivity in capturing the requirements for the decision analysis, which may threaten the construct. In addition, the path we followed in the selection of the decision mechanism was a situation that would significantly affect the output of our study. In order to address this situation, we used MCDA methodology by following the steps of a previous study [13] which did a similar work in a different field. Since the data was not in a simple structure and it contained different types and deficiencies, we went for a unified solution by adding fuzzy set calculations to the use of MCDA.

Conclusion validity:

Conclusion validity focuses on the reliability of our study's results if applied by other researchers. In order to ensure this validity type, we explained the methodology we used and our methods of obtaining information by detailing the process we followed. With this explanation, we believe that other researchers will be able to apply the approach and achieve similar results.

External validity:

External validity concerns the applicability of the outcome data by generalizing it beyond the scope of our study. With the approach proposed, we suggest the most appropriate method to the estimator who will do effort estimation. However, this choice will be between the models we have determined so far. It should be considered that there are other methods in addition to the chosen ones as the result of our literature review. In addition, we cannot say that the proposed method will be correct in all cases as the requirements and expectations may change, e.g., in the later stages of a same project or among different projects or organizations. We think that this study will become stronger by planning and realizing further empirical investigations.

6. CONCLUSION

6.1. Conclusion

In this study, an approach has been proposed for selecting the most suitable software effort estimation method considering the project characteristics and the needs of the stakeholders. The approach aims to assist users in choosing the most suitable estimation method in the targeted estimation context.

We started this study by identifying the distinctive criteria for software effort estimation methods. To identify these criteria, we used findings of several literature reviews and followed the approach of a similar study in [13]. Then, using the criteria, we prepared an expert-opinion survey to take the opinions of the experts in SEE. The aim of the expert-opinion survey was to enable us to establish a relationship between methods and criteria, in the form of a decision matrix. We calculated the rating values for the questions derived from the criteria using the answers given by eight experts to the survey.

Then, a questionnaire was prepared to be answered by the user (estimator) who wants to perform the estimation. The user would be able to see the accuracy scores of the estimation methods on the decision matrix by answering the questionnaire.

To make our work understandable, we explained it through an example evaluation based on the ISBSG dataset, and found that Neural Network and Case Based Reasoning are the most suitable methods in our estimation context. This method selection was partially supported with the results of the studies in the literature, and there appeared a need for further studies to validate the results of the evaluation approach.

We think that one of the most important factors determining the success of our approach is the number of experts who answered the expert opinion questionnaire. It would be good to add the industry experience as an added value to the study by sending our survey to the experts in the industry.

To advance this work one step further, we prepared a two-stage decision mechanism instead of the decision matrix. In this mechanism, the answers obtained as a result of the questionnaire answered by the estimator first pass the preselection in the decision tree and then choose the most appropriate method with the Fuzzy TOPSIS calculation. We prepared a multiple-case study for the validation of our proposal. With this study, we observed that our approach is valid and usable. In addition, by enabling the use of our approach with a tool implementation, we provided estimators to perform the estimation process in an easy way without knowing the details of the decision tree and fuzzy calculation. Also, we ensured that our approach was used by an external estimator who is an expert in the field, and we initially found with the feedback received that the tool we prepared was easy to use and understandable for carrying out software effort estimations.

6.2. Limitations

The proposed approach addresses the problem of selecting a suitable software effort estimation method through structuring the information and suggestions of the studies in literature into a decision approach. It will be beneficial to expand the scope of the work by including the gains in the field of effort estimation in software industry. For this purpose, it will be beneficial to include the opinions of the experts working in this field in the industry to the expert opinions analyzed within the scope of our study. In this way, in addition to the observed effects in academy, the effects experienced in industry can be reflected to the process of selecting a suitable estimation method. In addition, eight effort estimation methods, which are widely referenced in the literature, are analyzed within the scope of this study. Similarly, the scope of the study can be expanded by analyzing the effort estimation methods and features that are commonly used in the industry.

The most important factor affecting the selection of the appropriate estimation method is the answers to the estimator questions. Therefore, in order to answer the questionnaire, it is necessary to have sufficient knowledge of the characteristics of the project and related data to be included in the estimation. Failure to reflect the project characteristic to the decision matrix through estimator questions will negatively affect the selection of the estimation method. This situation may lead to poor estimation by choosing an unsuitable method. Our approach does not control whether the project characteristics are accurately reflected in the decision matrix. The responsibility in this matter is on the person who will perform the estimation.

Lastly, the majority of the experts who answered the expert-opinion survey are from the university. In future studies, reaching the experience of the people working in this field in the industry will increase the value of the expert-opinion survey results.

6.3. Future Work

Improvements can be made in the visualization of the tool interface prepared in the continuation of this study. In addition, we can invite more external estimators to use the tool, and the tool can be improved with the feedback we receive. Finally, to strengthen the evidence of validation, further empirical investigations can be designed and implemented.

7. REFERENCES

[1] Shepperd, Martin & Cartwright, Michelle. Predicting with sparse data. IEEE Trans. Softw.Eng., 27(11): 987–998, 2001

[2] Idri, Ali & Mbarki, Samir & Abran, Alain. (2004). Validating and understanding software cost estimation models based on neural networks. 433 - 434. 10.1109/ICTTA.2004.1307817.

[3] Nayebi, Fatih & Abran, Alain & Desharnais, Jean-Marc. (2015). Automated selection of a software effort estimation model based on accuracy and uncertainty. Artificial Intelligence Research. 4. 10.5430/air.v4n2p45.

[4] Sehra, Sumeet Kaur & Brar, Yadwinder & Kaur, Navdeep. (2012). Multi Criteria Decision Making Approach for Selecting Effort Estimation Model. International Journal of Computer applications. 39. 975-8887. 10.5120/4783-6989.

[5] Bansal, Ashu & Kumar, Brijesh & Garg, Rakesh. (2017). Multi-criteria decision making approach for the selection of software effort estimation model. Management Science Letters. 7. 285-296. 10.5267/j.msl.2017.3.003.

[6] Wen, Jianfeng & Li, Shixian & Lin, Zhiyong & Hu, Yong & Huang, Changqin. (2012). Systematic literature review of machine learning based software development effort estimation models. Information & Software Technology. 54. 41-59. 10.1016/j.infsof.2011.09.002.

[7] Jorgensen, M. & Shepperd, M. (2007). A Systematic Review of Software Development Cost Estimation Studies. IEEE Transactions on Software Engineering, 33, 33--53. doi: 10.1109/TSE.2007.256943.

[8] Marco, R. & Suryana, N. & Ahmad, S.S.S. (2019). A systematic literature review on methods for software effort estimation. Journal of Theoretical and Applied Information Technology. 97. 434-464.

[9] Idri, A., Amazal, F. A. & Abran, A. (2015). Analogy-based software development effort estimation: A systematic mapping and review. Inf. Softw. Technol., 58, 206-230.

[10] Bilgaiyan, Saurabh & Sagnika, Santwana & Mishra, Samaresh & Das, M.N. (2017). A Systematic Review on Software Cost Estimation in Agile Software Development. JOURNAL OF ENGINEERING SCIENCE AND TECHNOLOGY REVIEW. 10. 51-64. 10.25103/jestr.104.08.

[11] Shekhar, Shivangi & Kumar, Umesh. (2016). Review of Various Software Cost Estimation Techniques. International Journal of Computer Applications. 141. 31-34. 10.5120/ijca2016909867.

[12] Chirra, Sai Mohan Reddy & Reza, Hassan. (2019). A Survey on Software Cost Estimation Techniques. Journal of Software Engineering and Applications. 12. 10.4236/jsea.2019.126014.

[13] Ozakinci, Rana & Tarhan, Ayça. (2019). An Evaluation Approach for Selecting Suitable Defect Prediction Method at Early Phases. 10.1109/SEAA.2019.00040.

[14] A. B. Nassif, M. Azzeh, L. F. Capretz and D. Ho, "A comparison between decision trees and decision tree forest models for software development effort estimation," 2013 Third International Conference on Communications and Information Technology (ICCIT), 2013, pp. 220-224, doi: 10.1109/ICCITechnology.2013.6579553.

[15] Quinlan, J. R. (1990). Decision Trees and Decision-making. IEEE Transactions on Systems, Man and Cybernetics, 20(2), 339–346. https://doi.org/10.1109/21.52545.

[16] W. Pedrycz and Z. A. Sosnowski, "The design of decision trees in the framework of granular data and their application to software quality models," Fuzzy Sets and Systems, Vol. 1234, 2001, pp. 271-290.

[17] H. Berger, D. Merkl, and M. Dittenbach, "Exploiting Partial Decision Trees for Feature Subset Selection in e-Mail Categorization," in Proceedings of the 2006 ACM Symposium on Applied Computing (SAC 2006), Dijon, France, 2006, pp. 1105-1109.

[18] Figueira, José & Greco, Salvatore & Ehrgott, Matthias. (2005). Multiple Criteria Decision Analysis, State of the Art Surveys. Multiple Criteria Decision Analysis: State of the Art Surveys. 78. 10.1007/b100605.

[19] Belton, Valerie & Stewart, Theodor. (2002). Multiple Criteria Decision Analysis: An Integrated Approach. 10.1007/978-1-4615-1495-4.

[20] Saaty, R.W. (1987) The Analytic Hierarchy Process—What It Is and How It Is Used. Mathematical Modelling, 9, 161-176. http://dx.doi.org/10.1016/0270-0255(87)90473-8.

[21] Hwang, C.L. & Yoon, K. (1981). Multiple Attribute Decision Making: Methods and Applications. Springer-Verlag, New York. 10.1007/978-3-642-48318-9.

[22] Brans, J.P. & Mareschal, Bertrand. (2005). Chapter 5: PROMETHEE methods. Multiple Criteria Decision Analysis: State of the Art Surveys. 164-189.

[23] Figueira, J. & Mousseau, V. & Roy, B. (2005) Electre Methods. In: Multiple Criteria Decision Analysis: State of the Art Surveys. International Series in Operations Research & Management Science, vol 78. Springer, New York, NY.

[24] Multi-Criteria Decision Analysis, https://projects.ncsu.edu/nrli/decision-making/MCDA.php.

[25] Palczewski, Krzysztof & Sałabun, Wojciech. (2019). The fuzzy TOPSIS applications in the last decade. Procedia Computer Science. 159. 2294-2303. 10.1016/j.procs.2019.09.404.

[26] Hwang CL, Yoon KP. Multiple Attribute Decision Making: Methods and Applications, A State-of-the-Art Survey. Berlin: Springer Verlang; 1981.

[27] Sorin, Nadaban & Dzitac, Simona & Dzitac, Ioan. (2016). Fuzzy TOPSIS: A General View. Procedia Computer Science. 91. 823-831. 10.1016/j.procs.2016.07.088.

[28] Nădăban, S., Dzitac, S., & Dzitac, I. (2016). Fuzzy TOPSIS: A General View. Procedia Computer Science, 91(December 2016), 823–831. https://doi.org/10.1016/j.procs.2016.07.088

[29] ISBSG, International Software Benchmarking standards Group, http://www.isbsg.org.

[30] Plackett, R. L. "Karl Pearson and the Chi-Squared Test." International Statistical Review / Revue Internationale de Statistique 51, no. 1 (1983): 59–72. https://doi.org/10.2307/1402731.

[31] Vera, Tomas & Ochoa, Sergio & Perovich, Daniel. (2018). Survey of Software Development Effort Estimation Taxonomies. 10.13140/RG.2.2.14599.29601.

[32] Ballı, Serkan & Karasulu, Bahadır. (2013). Parallel Computing in Fuzzy Decision Making Systems. Pamukkale University Journal of Engineering Sciences. 19. 61-67. 10.5505/pajes.2013.91300.

[33] Venkataiah, V. & Mohanty, Rama & Nagaratna, M. (2017). Review on intelligent and soft computing techniques to predict software cost estimation. International Journal of Applied Engineering Research. 12. 12665-12681.

[34] Moosavi, S. H. Samareh & Bardsiri, V. Khatibi (2017). Satin bowerbird optimizer: A new optimization algorithm to optimize ANFIS for software development effort estimation. Eng. Appl. Artif. Intell., vol. 60. https://doi.org/10.1016/j.engappai.2017.01.006.

[35] Pospieszny, Przemyslaw & Czarnacka-Chrobot, Beata & Kobyliński, Andrzej. (2017). An effective approach for software project effort and duration estimation with machine learning algorithms. Journal of Systems and Software. 137. 10.1016/j.jss.2017.11.066.

[36] Azzeh, Mohammad & Neagu, Daniel & Cowling, Peter. (2011). Analogy-based software effort estimation using Fuzzy numbers. Journal of Systems and Software. 84. 270-284. 10.1016/j.jss.2010.09.028.

[37] Azzeh, Mohammad & Neagu, Daniel & Cowling, Peter. (2010). Fuzzy grey relational analysis for software effort estimation. Empirical Software Engineering. 15. 60-90. 10.1007/s10664-009-9113-0.

[38] Papatheocharous, Efi & Papadopoulos, Harris & Andreou, Andreas. (2012). Feature Subset Selection for Software Cost Modelling and Estimation. Engineering Intelligent Systems. 18.

[39] Nassif, Ali & Azzeh, Mohammad & Capretz, Luiz & Ho, Danny. (2016). Neural network models for software development effort estimation: a comparative study. Neural Computing and Applications. 27. 10.1007/s00521-015-2127-1.

[40] González-Ladrón-De-Guevara, F.; Fernández-Diego, M.; Lokan, C. (2016). The usage ofISBSG data fields in software effort estimation: A systematic mapping study. Journal of Systems and Software. 113:188-215. doi:10.1016/j.jss.2015.11.040.

[41] Bosu, Michael & MacDonell, Stephen. (2020). Experience: Quality Benchmarking of Datasets Used in Software Effort Estimation.

[42] Frank, E., Mark A., H., & Ian H., W. (2016). The WEKA Workbench. Online Appendix for “Data Mining: Practical Machine Learning Tools and Techniques.” Morgan Kaufmann, Fourth Edition. https://www.cs.waikato.ac.nz/ml/weka/.

[43] http://neuroph.sourceforge.net/about_project.html.

[44] https://zenodo.org/badge/latestdoi/373115654.

[45] Yin, R. K. (2017). Case Study Research and Applications: Design and Methods 6th Edition (6th ed., Issue 6e). Thousand Oaks, California: SAGE Publications, Inc., [2018] ©2018.

[46] Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2012). Experimentation in software engineering. Springer Publishing Company, Incorporated. https://doi.org/10.1007/978-3-642-29044-2

APPENDICES

Appendix 1 - Expert Opinion Survey

Expert Opinion Study on SEE Method Selection

This study is carried out by Duygu Deniz Erhan, an MSc. student at the Computer Engineering Department of Hacettepe University. This form is intended to inform you about the research conditions.

What is the purpose of the study?
The aim of the study is to assess the criteria that should be considered in the decision process of the most suitable method for software effort estimation (SEE).

What do we request from you?
If you agree to participate in the research, you are expected to complete the questionnaire by rating the related criteria on each SEE method. The survey is expected to take about 15 minutes. To reach for a timely conclusion of the study, we will kindly request 10-days response time.

Important note:
For questions in Section 3 to 8, we ask you to mark the methods that you are familiar with or have an expertise. Therefore, you may leave a question empty for the methods that you do not know or are unsure of.

How do we use the information you provide?
In the survey, you will be asked to identify your title and organization type, only for statistical analysis. Your answers will be kept strictly confidential and will only be evaluated by the administrators. The information to be obtained from the participants will be evaluated collectively and used in the MSc. study by Duygu Deniz Erhan. The data you provide will not be matched with the credentials collected in the forms.

For more information, you may contact Duygu Deniz Erhan with the administrator information given below.

Thank you in advance for your participation in this study.

* Required

1. If you read the above information and accept the terms of agreement, click I agree to continue. *

Mark only one oval.

- I agree
- I don't want to participate   *Skip to question 26*

Administrator Information

Duygu DENİZ ERHAN
E-mail (personal): duygudeniz06@gmail.com
E-mail (university): n16222537@cs.hacettepe.edu.tr
Telephone: +90 505 4277160

Expert Opinion Study on SEE Method Selection

Expert Information

Please fill in the below personal information. Note that personal info will not be published anywhere. It will only processed for descriptive statistics anonymously.

2. Name *

____

3. Email adress *

____

4. Organization Type *

Mark only one oval.

- ◯ University
- ◯ Private Company
- ◯ Government
- ◯ Other: _______________

5. Title *

____

6. 1) What is your degree of knowledge in SEE? *

Mark only one oval.

	1	2	3	4	5	
Very Low	◯	◯	◯	◯	◯	Very High

7. 2) How many years of experience do you have in the area of SEE? *

Mark only one oval.

- ◯ No experience
- ◯ Less than 3 years
- ◯ 3 - 5 years
- ◯ 6 - 10 years
- ◯ 11 - 20 years
- ◯ More than 20 years
- ◯ Other: ______________________

8. 3) What is your degree of knowledge in estimation methods in general? *

Mark only one oval.

	1	2	3	4	5	
Very Low	◯	◯	◯	◯	◯	Very High

9. 4) Which of the following methods do you consider yourself an expert? *

Mark only one oval per row.

	No experience	I am familiar with this method	I have expertise in this method
Expert Judgement	⬭	⬭	⬭
Analogy Based	⬭	⬭	⬭
Neural Networks	⬭	⬭	⬭
Bayesian Networks	⬭	⬭	⬭
Linear Regression	⬭	⬭	⬭
Decision Trees	⬭	⬭	⬭
Case-Base Reasoning	⬭	⬭	⬭
Support Vector Regression	⬭	⬭	⬭

Model Construction

SEE model constructing approaches is investigated under this group.

*** Please feel free to leave your answer for a method empty if you are not familiar with it. ***

10. 5) Please select the convenient option on “Approach to Construct the SEE Model” with the below methods.

Mark only one oval per row.

	Dependent on data	Based on human judgement	Can adress both
Expert Judgement	⬭	⬭	⬭
Analogy Based	⬭	⬭	⬭
Neural Networks	⬭	⬭	⬭
Bayesian Networks	⬭	⬭	⬭
Linear Regression	⬭	⬭	⬭
Decision Trees	⬭	⬭	⬭
Case-Base Reasoning	⬭	⬭	⬭
Support Vector Regression	⬭	⬭	⬭

Data Characteristics

There are several characteristics which are crucial to address the constraints of the data that will be used for building the SEE model: type of input data, dataset size, number of parameters.

*** Please feel free to leave your answer for a method empty if you are not familiar with it. ***

11. 6) Please select the convenient option on "Type of input data" of the below methods in SEE.

Mark only one oval per row.

	Categorical	Numerical	Can adress both
Expert Judgement	⬭	⬭	⬭
Analogy Based	⬭	⬭	⬭
Neural Networks	⬭	⬭	⬭
Bayesian Networks	⬭	⬭	⬭
Linear Regression	⬭	⬭	⬭
Decision Trees	⬭	⬭	⬭
Case-Base Reasoning	⬭	⬭	⬭
Support Vector Regression	⬭	⬭	⬭

12. 7) What do you think about the minimum "dataset size" required for building/training an SEE model with the below methods?

Mark only one oval per row.

	No data required	Small	Medium	Large	Very Large
Expert Judgement	⬭	⬭	⬭	⬭	⬭
Analogy Based	⬭	⬭	⬭	⬭	⬭
Neural Networks	⬭	⬭	⬭	⬭	⬭
Bayesian Networks	⬭	⬭	⬭	⬭	⬭
Linear Regression	⬭	⬭	⬭	⬭	⬭
Decision Trees	⬭	⬭	⬭	⬭	⬭
Case-Base Reasoning	⬭	⬭	⬭	⬭	⬭
Support Vector Regression	⬭	⬭	⬭	⬭	⬭

13. 8) Is the number of parameters regarding past project data an indicator of building/training an SEE model with the below methods ?

Mark only one oval per row.

	Yes	No
Expert Judgement	◯	◯
Analogy Based	◯	◯
Neural Networks	◯	◯
Bayesian Networks	◯	◯
Linear Regression	◯	◯
Decision Trees	◯	◯
Case-Base Reasoning	◯	◯
Support Vector Regression	◯	◯

Data quality

The quality characteristics of the data to be used to construct the SEE model are discussed under this group: uncertainty, missing values, outliers.

*** Please feel free to leave your answer for a method empty if you are not familiar with it. ***

14. 9) To what extend do you think the following methods can handle "uncertainty" in SEE data?

Uncertainty is the degree to which data is inaccurate, imprecise, untrusted or unknown.

Mark only one oval per row.

	Very Low	Low	Average	High	Very High
Expert Judgement					
Analogy Based					
Neural Networks					
Bayesian Networks					
Linear Regression					
Decision Trees					
Case-Base Reasoning					
Support Vector Regression					

15. 10) To what extend do you think the following methods can handle "missing values" in SEE data?

Missing data for certain variables can lead to poor estimations in some sensitive models.

Mark only one oval per row.

	Very Low	Low	Average	High	Very High
Expert Judgement					
Analogy Based					
Neural Networks					
Bayesian Networks					
Linear Regression					
Decision Trees					
Case-Base Reasoning					
Support Vector Regression					

14. 9) To what extend do you think the following methods can handle "uncertainty" in SEE data?

Uncertainty is the degree to which data is inaccurate, imprecise, untrusted or unknown.

Mark only one oval per row.

	Very Low	Low	Average	High	Very High
Expert Judgement	⬭	⬭	⬭	⬭	⬭
Analogy Based	⬭	⬭	⬭	⬭	⬭
Neural Networks	⬭	⬭	⬭	⬭	⬭
Bayesian Networks	⬭	⬭	⬭	⬭	⬭
Linear Regression	⬭	⬭	⬭	⬭	⬭
Decision Trees	⬭	⬭	⬭	⬭	⬭
Case-Base Reasoning	⬭	⬭	⬭	⬭	⬭
Support Vector Regression	⬭	⬭	⬭	⬭	⬭

15. 10) To what extend do you think the following methods can handle "missing values" in SEE data?

Missing data for certain variables can lead to poor estimations in some sensitive models.

Mark only one oval per row.

	Very Low	Low	Average	High	Very High
Expert Judgement	⬭	⬭	⬭	⬭	⬭
Analogy Based	⬭	⬭	⬭	⬭	⬭
Neural Networks	⬭	⬭	⬭	⬭	⬭
Bayesian Networks	⬭	⬭	⬭	⬭	⬭
Linear Regression	⬭	⬭	⬭	⬭	⬭
Decision Trees	⬭	⬭	⬭	⬭	⬭
Case-Base Reasoning	⬭	⬭	⬭	⬭	⬭
Support Vector Regression	⬭	⬭	⬭	⬭	⬭

16. 11) To what extend do you think the following methods can handle "outliers" in SEE data?

An outlier is an observation that lies an abnormal distance from other values in a dataset.

Mark only one oval per row.

	Very Low	Low	Average	High	Very High
Expert Judgement	⬭	⬭	⬭	⬭	⬭
Analogy Based	⬭	⬭	⬭	⬭	⬭
Neural Networks	⬭	⬭	⬭	⬭	⬭
Bayesian Networks	⬭	⬭	⬭	⬭	⬭
Linear Regression	⬭	⬭	⬭	⬭	⬭
Decision Trees	⬭	⬭	⬭	⬭	⬭
Case-Base Reasoning	⬭	⬭	⬭	⬭	⬭
Support Vector Regression	⬭	⬭	⬭	⬭	⬭

Method characteristics

The characteristics of the methods to be used to construct the SEE model are discussed under this group: interpretable, easy to use (not complex), speedy, maintainable, adaptive.

*** Please feel free to leave your answer for a method empty if you are not familiar with it. ***

17. 12) To what extent do you think the following methods are "interpretable" by its users in SEE?

Interpretability is the degree of which the user can understand the cause of any result (output).

Mark only one oval per row.

	Very Low	Low	Average	High	Very High
Expert Judgement	◯	◯	◯	◯	◯
Analogy Based	◯	◯	◯	◯	◯
Neural Networks	◯	◯	◯	◯	◯
Bayesian Networks	◯	◯	◯	◯	◯
Linear Regression	◯	◯	◯	◯	◯
Decision Trees	◯	◯	◯	◯	◯
Case-Base Reasoning	◯	◯	◯	◯	◯
Support Vector Regression	◯	◯	◯	◯	◯

18. 13) To what extent do you think the following methods are "easy to use (not complex)" for SEE?

Ease of use (not being complex) is the degree of which the method is not complicated in design.

Mark only one oval per row.

	Very Low	Low	Average	High	Very High
Expert Judgement	◯	◯	◯	◯	◯
Analogy Based	◯	◯	◯	◯	◯
Neural Networks	◯	◯	◯	◯	◯
Bayesian Networks	◯	◯	◯	◯	◯
Linear Regression	◯	◯	◯	◯	◯
Decision Trees	◯	◯	◯	◯	◯
Case-Base Reasoning	◯	◯	◯	◯	◯
Support Vector Regression	◯	◯	◯	◯	◯

19. 14) To what extent do you think the following methods can be used to build SEE models in a short time? (i.e. How speedy are they in model building and execution?)

Speed is the degree of which the method is build in a short time and performs fast in general.

Mark only one oval per row.

	Very Low	Low	Average	High	Very High
Expert Judgement					
Analogy Based					
Neural Networks					
Bayesian Networks					
Linear Regression					
Decision Trees					
Case-Base Reasoning					
Support Vector Regression					

20. 15) To what extent do you think the following methods are "maintainable" when used for SEE?

Maintability is the degree of which the method is easy to manage in time.

Mark only one oval per row.

	Very Low	Low	Average	High	Very High
Expert Judgement	◯	◯	◯	◯	◯
Analogy Based	◯	◯	◯	◯	◯
Neural Networks	◯	◯	◯	◯	◯
Bayesian Networks	◯	◯	◯	◯	◯
Linear Regression	◯	◯	◯	◯	◯
Decision Trees	◯	◯	◯	◯	◯
Case-Base Reasoning	◯	◯	◯	◯	◯
Support Vector Regression	◯	◯	◯	◯	◯

21. 16) Do you think the following methods are "adaptive" for new data in SEE?

Being adaptive means the method can accept new data without re-running the SEE model.

Mark only one oval per row.

	Yes	No
Expert Judgement	◯	◯
Analogy Based	◯	◯
Neural Networks	◯	◯
Bayesian Networks	◯	◯
Linear Regression	◯	◯
Decision Trees	◯	◯
Case-Base Reasoning	◯	◯
Support Vector Regression	◯	◯

Project context	The factors related to the context information of the project subject to SEE are discussed under this group: iteration, domain, size *** Please feel free to leave your answer for a method empty if you are not familiar with it. ***

22. 17) Do you think that iteration in software development life cycle is an affecting factor in SEE with the following methods?

Mark only one oval per row.

	Yes	No
Expert Judgement	◯	◯
Analogy Based	◯	◯
Neural Networks	◯	◯
Bayesian Networks	◯	◯
Linear Regression	◯	◯
Decision Trees	◯	◯
Case-Base Reasoning	◯	◯
Support Vector Regression	◯	◯

23. 18) Do you think that "domain" information of software project is an affecting factor in SEE with the following methods?

Mark only one oval per row.

	Yes	No
Expert Judgement	◯	◯
Analogy Based	◯	◯
Neural Networks	◯	◯
Bayesian Networks	◯	◯
Linear Regression	◯	◯
Decision Trees	◯	◯
Case-Base Reasoning	◯	◯
Support Vector Regression	◯	◯

24. 19) Do you think that "size" information of software project is an affecting factor in SEE with the following methods?

Mark only one oval per row.

	Yes	No
Expert Judgement	◯	◯
Analogy Based	◯	◯
Neural Networks	◯	◯
Bayesian Networks	◯	◯
Linear Regression	◯	◯
Decision Trees	◯	◯
Case-Base Reasoning	◯	◯
Support Vector Regression	◯	◯

25. 20) To what extent do you think the following methods are useful in SEE for single project vs cross-project estimation?

Mark only one oval per row.

	Single-project	Cross-project	Can adress both
Expert Judgement	◯	◯	◯
Analogy Based	◯	◯	◯
Neural Networks	◯	◯	◯
Bayesian Networks	◯	◯	◯
Linear Regression	◯	◯	◯
Decision Trees	◯	◯	◯
Case-Base Reasoning	◯	◯	◯
Support Vector Regression	◯	◯	◯

Comments/ Feedback

26. Please specify any other criteria that you suggest to consider when choosing SEE method, with your rationale (e.g. the effect of these criteria on the methods).

27. Please specify on any other method that you suggest to include in SEE method selection.

28. Do you have any comments or further suggestions to improve our expert opinion study?

Appendix 2 - Effort Estimation Method and Tool Feedback Survey

Effort Estimation Method and Tool Feedback

This study is carried out by Duygu Deniz Erhan, an MSc. student at the Computer Engineering Department of Hacettepe University. This form is intended to inform you about the feedback.

We asked you to do effort estimation model selection using the method(Decision Matrix) and tool prepared. With this survey, we aim to receive your feedback on the use of method and tool.
The survey is expected to take about 5 minutes. Thank you for your valuable feedback.

For more information, you may contact Duygu Deniz Erhan with the administrator information given below.

Thank you in advance for your participation in this study.

* Required

If you read the above information and accept the terms of agreement, click I agree to continue. *

- ○ I agree
- ○ I don't want to participate

Administrator Information

Duygu DENİZ ERHAN
E-mail (personal): duygudeniz06@gmail.com
E-mail (university): n16222537@cs.hacettepe.edu.tr
Telephone: +90 505 4277160

Effort Estimation Method and Tool Feedback

* Required

User Information

Name *

Your answer

About Method Usage

Here are questions about the effort estimation method selection study using the Decision Matrix prepared.

To what extent do you think the use of the method in terms of comprehensibility? *

	1	2	3	4	5	
Very Low	○	○	○	◉	○	Very High

To what extent do you think the use of the method in terms of ease of use? *

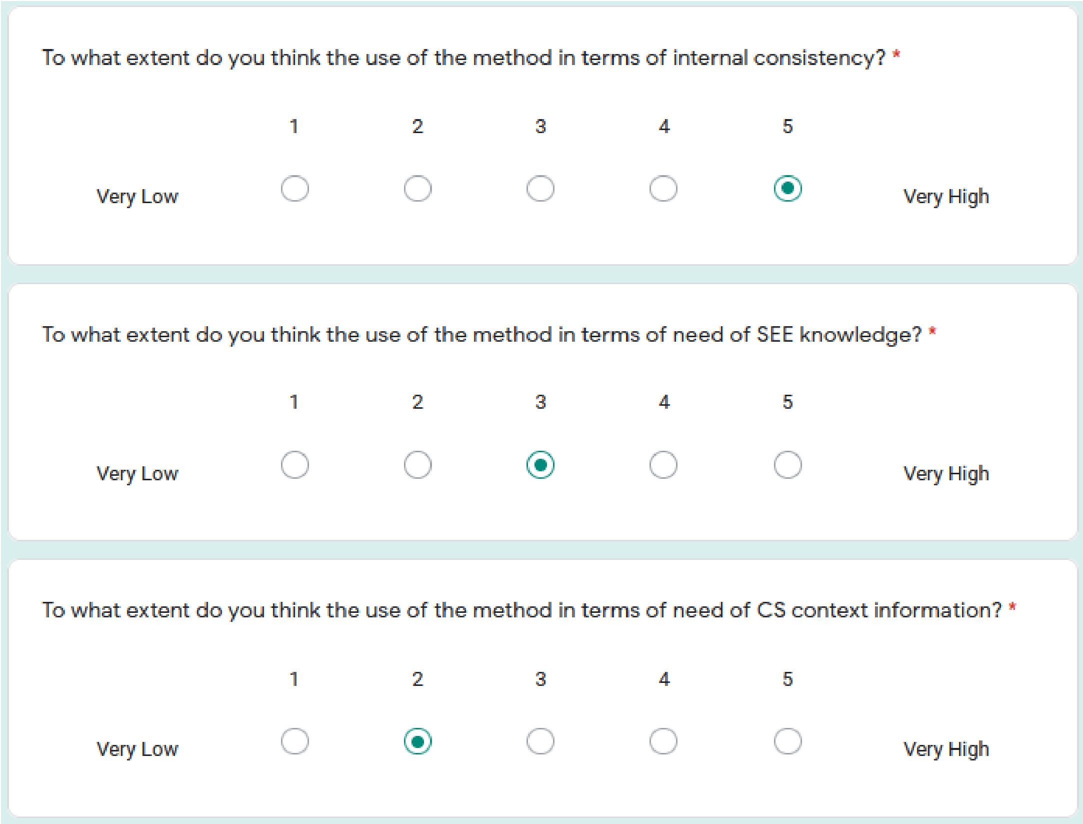
	1	2	3	4	5	
Very Low	○	○	○	◉	○	Very High

To what extent do you think the use of the method in terms of simplicity? *

	1	2	3	4	5	
Very Low	○	○	○	○	◉	Very High

To what extent do you think the use of the method in terms of clearness? *

	1	2	3	4	5	
Very Low	○	○	○	◉	○	Very High



To what extent do you think the use of the method in terms of internal consistency? *

	1	2	3	4	5	
Very Low						Very High

To what extent do you think the use of the method in terms of need of SEE knowledge? *

	1	2	3	4	5	
Very Low						Very High

To what extent do you think the use of the method in terms of need of CS context information? *

	1	2	3	4	5	
Very Low						Very High

About Tool Usage

Here are questions about the effort estimation method selection study using the Decision Matrix prepared.

To what extent do you think the use of the method in terms of function understandability? *

	1	2	3	4	5	
Very Low	○	○	○	○	◉	Very High

To what extent do you think the use of the method in terms of ease of use? *

	1	2	3	4	5	
Very Low	○	○	○	○	◉	Very High

To what extent do you think the use of the method in terms of simplicity? *

	1	2	3	4	5	
Very Low	○	○	○	◉	○	Very High

To what extent do you think the use of the method in terms of fairness of GUI design ? *

	1	2	3	4	5	
Very Low	○	○	○	◉	○	Very High

To what extent do you think the use of the method in terms of operability ? *

	1	2	3	4	5	
Very Low	○	○	○	◉	○	Very High

Comments/ Feedback

Do you have any comments or further suggestions to improve our form and our study?

Your answer

Appendix 3 - Papers Derived From Thesis

D.D.Erhan, A.Kolukısa-Tarhan, and R.Özakıncı, "Selecting Suitable Software Effort Estimation Method", in proceedings of the 30th International Workshop on Software Measurement (IWSM) and the 15th International Conference on Software Process and Product Measurement (MENSURA) (CEUR Proceedings, Vol. 2725, Paper11).