Web Effort Estimation: the Value of Cross-company Data Set Compared to Single-company Data Set

Filomena Ferrucci
University of Salerno
Via Ponte don Melillo
Fisciano (SA), Italy
+39 089963374
fferrucci@unisa.it

Emilia Mendes
Zayed University
Dubai Campus, P.O. Box 19282
Dubai, UAE
+971558073407
emilia.mendes@zu.ac.ae

Federica Sarro
University of Salerno
Via Ponte don Melillo
Fisciano (SA), Italy
+39 089963323
fsarro@unisa.it

ABSTRACT
This study investigates to what extent Web effort estimation models built using cross-company data sets can provide suitable effort estimates for Web projects belonging to another company, when compared to Web effort estimates obtained using that company's own data on their past projects (single-company data set). It extends a previous study (S3) where these same research questions were investigated using data on 67 Web projects from the Tukutuku database. Since S3 was carried out, data on other 128 Web projects was added to Tukutuku; therefore this study uses the entire set of 195 projects from the Tukutuku database, which now also includes new data from other single-company data sets. Predictions between cross-company and single-company models are compared using Manual Stepwise Regression+Linear Regression and Case-Based Reasoning. In addition, we also investigated to what extent applying a filtering mechanism to cross-company datasets prior to building prediction models can affect the accuracy of the effort estimates they provide. The present study corroborates the conclusions of S3 since the cross-company models provided much worse predictions than the single-company models. Moreover, the use of the filtering mechanism significantly improved the prediction accuracy of cross-company models when estimating single-company projects, making it comparable to that using single-company datasets.

Categories and Subject Descriptors
D.2.9 [Software Engineering]: Management---Cost estimation; D.2.8 [Software Engineering]: Metrics---Process metrics, Product metrics

General Terms
Management, Measurement, Experimentation.

Keywords
Cross-company effort model, single-company effort model, cost estimation, effort estimation, filtering mechanism, stepwise regression, case-based reasoning, Web projects.

PROMISE'12, September 21-22, 2012 Lund, Sweden

1. INTRODUCTION
When planning a project, the estimation of development effort/cost is a critical management activity, also crucial for the competitiveness of a software company. It aims at predicting an accurate effort estimate and using this information to allocate resources adequately, such that projects are completed within time and on budget. Most research in this field has looked at improving the estimation process via the use of past data from finished projects to build formal estimation models to provide effort predictions for new projects [9]. However, there are issues that a company faces that are associated with building its own data set of past projects, which are the following:

i) The time required to accumulate enough data on past projects from a single company may be prohibitive.
ii) By the time the data set is large to be of use, technologies used by the company may have changed, and older projects may no longer be representative of current practices.
iii) Care is necessary as data needs to be collected in a consistent manner.

These three problems have motivated previous studies to investigate to what extent effort estimation models built using cross-company (CC) data sets, i.e., data sets that contain project data volunteered by several companies, can provide suitable effort estimates for projects belonging to another company, when compared to effort estimates obtained using that company's own data on their past projects (single-company data set (SC)). A description and comparison of most of these studies is given in [9]. Some of these studies have focused solely on Web projects because, such as ourselves, they see Web and software development differing in a number of areas, such as: application characteristics, primary technologies used, approach to quality delivered, development process drivers, availability of the application, customers (stakeholders), update rate (maintenance cycles), people involved in development, architecture and network, disciplines involved, legal, social, and ethical issues, and information structuring and design. A detailed discussion on this issue is provided in [18].

The first study comparing cross-company to single-company predictions using Web project data, by Kitchenham and Mendes [8] (S1), investigated, using data on 53 Web projects (40 cross-company and 13 from a single-company), to what extent a cross-company cost model could be successfully employed to estimate development effort for single-company Web projects. Their effort models were built using Forward Stepwise Regression (SWR) and they found that cross-company predictions were significantly worse than single-company predictions. S1 was extended by

Mendes and Kitchenham [20] (S2), who used SWR and Case-based reasoning (CBR), and data on 67 Web projects from the Tukutuku database (53 cross-company and 14 from a single-company). They built two cross-company and one single-company models and found that both SWR cross-company models provided predictions significantly worse than the single company predictions, and CBR cross-company data provided predictions significantly better than the single company predictions. By 2007 another 83 projects had been volunteered to the Tukutuku database (68 cross-company and 15 from a single-company), and were used by Mendes et al. [21] to partially replicate S2 (only one cross-company model was built) (S3), using SWR and CBR. They corroborated some of S2's findings (SWR cross-company model provided predictions significantly worse than single-company predictions); however S2 found CBR cross-company predictions to be superior to CBR single-company predictions, which is the opposite of what we obtained in S3. Later, in 2008, Mendes et al. [22] (S4) extended S3 to fully replicate S2. They used the same dataset used in S3, and their results corroborated most of those obtained in S2. The main difference between S2 and S4 was that one of S4's SWR cross-company models showed similar predictions to the single-company model, which contradicts the findings from S2. After S4 was published, other 45 projects were volunteered to the Tukutuku dataset, therefore the objective and main contribution of this study is to extend S3, using the entire set of 195 projects from the Tukutuku dataset, which includes data from two single-company data sets. Predictions between cross-company and single-company models are compared using Manual Stepwise Regression + Linear Regression (MSWR+LR) and Case-Based Reasoning (CBR). In addition, we also investigated to what extent applying a filtering mechanism to cross-company datasets prior to building prediction models with MSWR+LR can affect the accuracy of the effort estimates they provide. This question has not been previously investigated in the context of Web effort estimation, thus making an additional research contribution of this work.

Note that our study is an extension of study S3, rather than an independent replication, because S3 used part of the Tukutuku data we are employing herein.

The remainder of this paper is organised as follows: Section 2 details the research method employed, followed by the description of how we built all the prediction models used herein in Section 3. Section 4 presents our results, followed by threats to validity in Section 5. Related work are reported in Section 6 and finally our conclusions are discussed in Section 7.

2. RESEARCH METHOD

2.1 RESEARCH QUESTIONS

This study addressed the same questions investigated by S1 [8], S2 [20], S3 [21], and S4 [22], as follows:

RQ1: How successful is a cross-company dataset at estimating effort for projects from a single company?

RQ2: How successful is the use of a cross-company dataset, compared to a single-company dataset, for effort estimation?

In addition, we also investigated the following research questions, which have not been previously investigated in the context of Web effort estimation, thus making an additional research contribution to this body of knowledge:

RQ3: How successful is a filtered cross-company dataset at estimating effort for projects from a single company?

RQ4: How successful is the use of a filtered cross-company dataset, compared to

a) the non-filtered cross-company dataset, and

b) a single-company dataset?

Given that CC datasets contain project data volunteered by a wide range of companies, one can argue that overall the CC projects are likely to be more heterogeneous than SC projects. Thus, our two additional research questions aim to investigate to what extent applying a filtering mechanism to CC datasets prior to building prediction models can affect the accuracy of the effort estimates they provide. As detailed in section 2.4.3, such filtering mechanism aims to create a more homogeneous training set by adding to it only CC projects that are judged to be more similar to those in the SC dataset, using the Euclidean distance as similarity measure.

2.2 Dataset Description

The analysis presented in this paper used the data from the 195 Web projects available in the Tukutuku database [18]. The data represents a wide range of Web applications, from static to dynamic applications often developed using content management systems. More detailed characteristics of this database are as follows:

- Projects came mostly from 10 different countries, mainly New Zealand (47%), Italy (17%), Spain (16%), Brazil (10%), United States (4%), England (2%), and Canada (2%).
- Project data has been volunteered by 51 different companies, where 111 of these projects come from six different single companies, which volunteered respectively 14, 20, 15, 13, 31, and 18 projects.
- Project types are new developments (65.6%) or enhancement projects (34.4%).
- About dynamic technologies, PHP is used in 42.6% of the projects, ASP (VBScript or .Net) in 13.8%, Perl in 11.8%, J2EE in 9.2%, while 9.2% of the projects used other solutions. The remaining projects used only HTML and/or Javascript.

Each Web project in the Tukutuku database is characterized by process and product variables [16], which are described in Table 1. These size measures and cost drivers have been obtained from the results of a survey investigation [18], using data from on-line Web forms aimed at giving quotes on Web development projects. In addition, these measures and cost drivers have also been confirmed by an established Web company and a second survey involving 33 Web companies in New Zealand. Consequently it is our belief that the 25 variables identified are measures that are meaningful to Web companies and are constructed from information their customers can provide at a very early stage in project development. Within the context of the Tukutuku project, a new high-effort feature/function employs at least 15 hours to be developed by one experienced developer, and a high-effort adapted feature/function employs at least 4 hours to be adapted by one experienced developer. These values are based on collected data.

As for data quality, Web companies that volunteered data for the Tukutuku database did not use any automated measurement tools for effort data collection. Therefore in order to identify guesstimates from more accurate effort data, we asked companies how their effort data was collected (see Table 2). At least for

93.8% of Web projects in the Tukutuku database, effort values were based on more than just guesstimates.

As in previous studies (S1, S2, S3, and S4), we excluded from our analysis some variables based on the following criteria:

- More than 50% of instances of a variable were zero.
- The variable was categorical (nominal and ordinal).
- The variable was related to another variable, in which case both could not be included in the same model. To measure the strength of the association between variables we used the Spearman's rank correlation statistical test.

In addition, also as in previous studies, the motivation to exclude categorical variables from our analysis was that the Tukutuku categorical variables had many levels, thus requiring a large number of dummy variables which rapidly reduce the degrees of freedom for analysis.

Tables 3 and 4 contain the summary statistics for the two single-company datasets (i.e., SC1 and SC2) and the cross-company data sets (i.e., CC1 and CC2) employed in our analysis. They suggest that there are both similarities and differences between the single- and cross-company projects. Indeed, single-company projects from Company 50 used about twice the number of languages as the cross-company projects, while single- and cross- company projects from Company 51 used in average a similar number of languages (i.e., 4). As for the size of the development teams, we can observe that the cross-company projects from company of both companies required a higher number of developers with respect to the single-company projects. However, cross-company developers presented, on average, less experience than single-company developers. Moreover, cross-company applications are bigger, in number of Web pages and images, than those of the single-company applications. And, therefore, the effort spent on cross-company projects is greater than that spent on single-company projects excepts for Company 50, where the effort spent on single-company projects is smaller than that spent on cross-company projects. This is probably due to the fact that in this case the number of high-effort features/functions required by the single-company applications is in average very high (i.e., 74.06) with respect to the one required by the cross-company ones (i.e., 2.69). These differences are not sufficient to suggest that the cross-company data cannot be useful to estimate effort for single company projects.

2.3 Evaluation Criteria

The accuracy of the effort estimates was assessed using statistical analysis together with several accuracy measures based on absolute residuals (i.e., unsigned difference between actual effort and estimated effort). To check whether the differences in estimation accuracy between the cross-company and single-company datasets were legitimate or due to chance, we employed the Wilcoxon Signed Ranks test for two related samples to check if both sets of absolute residuals came from the same population. We set $\alpha = 0.05$ [7][22]. Though it is important to assess whether an estimation model is statistically better than another, it is in addition crucial to assess the magnitude of the improvement. To this end we employed Cohen's *d* effect size, where results are considered small for $0.2 \leq d < 0.5$, medium for $0.5 \leq d < 0.8$, and large for $d \geq 0.8$ [3]. As for accuracy measures, we employed the Mean of the Absolute Residuals (MAR), the Mean Magnitude of Relative Error (MMRE), the Median MRE (MdMRE), and the Prediction at 25% (Pred(25)) [1].

MRE is the basis for calculating MMRE and MdMRE, and is defined as:

$$\text{MRE} = \frac{|e - \hat{e}|}{e} \quad (1)$$

where e represents actual effort and ê estimated effort. The difference between MMRE and MdMRE is that the former is sensitive to predictions containing extreme MRE values.

Pred(*n*) measures the percentage of estimates that are within *n*% of the actual values and *n* is usually set at 25%.

Table 1 - Variables for the Tukutuku database

Variable Name	Description
Country	Country company belongs to.
Established	Year when company was established.
nPeopleWD	Number of people who work on Web design and development.
TypeProj	Type of project (new or enhancement).
nLang	Number of different development languages used
DocProc	If project followed defined and documented process.
ProImpr	If project team involved in a process improvement programme.
Metrics	If project team part of a software metrics programme.
DevTeam	Size of a project's development team.
TeamExp	Average team experience with the development language(s) employed.
TotEff	Actual total effort used to develop the Web application.
EstEff	Estimated total effort necessary to develop the Web application.
Accuracy	Procedure used to record effort data.
TypeApp	Type of Web application developed.
TotWP	Total number of Web pages (new and reused).
NewWP	Total number of new Web pages.
TotImg	Total number of images (new and reused).
NewImg	Total number of new images created.
Fots	Number of features reused without any adaptation.
HFotsA	Number of reused high-effort features/functions adapted.
Hnew	Number of new high-effort features/functions.
TotHigh	Total number of high-effort features/functions
FotsA	Number of reused low-effort features adapted.
New	Number of new low-effort features/functions.
TotNHigh	Total number of low-effort features/functions

Table 2 - How effort data was collected

Data Collection Method	# of Projects	% of Projects
Hours worked per project task per day	81	41.5
Hours worked per project per day/week	40	20.5
Total hours worked each day or week	62	31.8
No timesheets (guesstimates)	12	6.2

Table 3. Summary statistics for variables in SC1 and CC1 datasets

Single-company data set SC1 – 31 projects					
	Mean	**Median**	**St.Dev.**	**Min**	**Max**
nLang	5.19	5.00	0.65	5.00	8.00
DevTeam	1.35	1.00	0.49	1.00	2.00
TeamExp	5.61	6.00	1.45	4.00	8.00
TotWP	42.61	18.00	54.78	3.00	200.00
NewWP	2.65	0.00	6.68	0.00	30.00
TotImg	26.13	4.00	78.03	0.00	405.00
NewImg	3.06	0.00	8.10	0.00	40.00
Fots	5.23	5.00	3.03	0.00	10.00
HFotsA	74.06	17.00	135.74	2.00	611.00
Hnew	0.00	0.00	0.00	0.00	0.00
TotHigh	74.06	17.00	135.74	2.00	611.00
FotsA	1.52	0.00	3.91	0.00	20.00
New	2.90	0.00	4.70	0.00	15.00
TotNHigh	4.42	0.00	8.06	0.00	35.00
TotEff	604.06	98	1109.39	16	3644
Cross-Company data set CC1 – 164 projects					
	Mean	**Median**	**St.Dev.**	**Min**	**Max**
nLang	3.64	4.00	1.43	1.00	8.00
DevTeam	2.81	2.00	2.52	1.00	23.00
TeamExp	3.49	3.00	1.95	1.00	10.00
TotWP	74.55	29.50	200.81	1.00	2000.00
NewWP	58.41	15.00	194.13	0.00	1980.00
TotImg	112.28	50.00	233.33	0.00	1820.00
NewImg	44.93	3.50	135.81	0.00	1000.00
Fots	2.80	0.00	6.62	0.00	63.00
HFotsA	0.22	0.00	0.64	0.00	4.00
Hnew	2.47	0.00	5.03	0.00	27.00
TotHigh	2.69	0.00	5.07	0.00	27.00
FotsA	2.38	1.00	4.64	0.00	38.00
New	4.49	1.00	10.31	0.00	99.00
TotNHigh	6.87	4.00	13.97	0.00	137.00
TotEff	442.41	84.75	904.22	1.10	5000.00

Table 4. Summary statistics for variables in SC2 and CC2 datasets

Single-company data set SC2 – 18 projects					
	Mean	**Median**	**St.Dev.**	**Min**	**Max**
nLang	4.00	4.00	0.00	4.00	4.00
DevTeam	1.06	1.00	0.24	1.00	2.00
TeamExp	5.00	5.00	0.00	5.00	5.00
TotWP	37.89	24.00	36.06	6.00	140.00
NewWP	15.61	11.50	13.81	0.00	40.00
TotImg	85.28	90.00	33.01	0.00	150.00
NewImg	16.94	15.00	15.35	0.00	40.00
Fots	7.61	1.00	15.31	0.00	63.00
HFotsA	0.11	0.00	0.47	0.00	2.00
Hnew	1.39	0.00	2.28	0.00	8.00
TotHigh	1.50	0.00	2.36	0.00	8.00
FotsA	7.72	5.00	10.52	0.00	38.00
New	20.78	14.00	24.15	0.00	99.00
TotNHigh	28.50	17.50	34.23	0.00	137.00
TotEff	163.06	160.00	112.92	30.00	480.00
Cross-Company data set CC2 – 177 projects					
	Mean	**Median**	**St.Dev.**	**Min**	**Max**
nLang	3.88	4.00	1.52	1.00	8.00
DevTeam	2.73	2.00	2.44	1.00	23.00
TeamExp	3.71	3.00	2.09	1.00	10.00
TotWP	72.69	27.00	194.35	1.00	2000.00
NewWP	53.00	10.00	187.68	0.00	1980.00
TotImg	99.94	25.00	228.99	0.00	1820.00
NewImg	40.44	1.00	131.45	0.00	1000.00
Fots	2.74	0.00	4.26	0.00	21.00
HFotsA	13.16	0.00	62.71	0.00	611.00
Hnew	2.15	0.00	4.88	0.00	27.00
TotHigh	15.31	1.00	62.46	0.00	611.00
FotsA	1.68	0.00	2.93	0.00	20.00
New	2.56	0.00	3.94	0.00	19.00
TotNHigh	4.24	3.00	4.99	0.00	35.00
TotEff	499.13	80.00	979.37	1.10	5000.00

2.4 Estimation and Filtering Techniques Employed

The techniques used to obtain effort estimates were Manual Stepwise Regression (MSWR) combined with Linear Regression (LR), and Case-Based Reasoning (CBR). Moreover, we employed the Nearest Neighbor (NN) Filtering technique, first suggested by Turhan et al. [2], to leave in the training set only the most similar projects to those in the validation set prior to build the regression based model. All results presented here were obtained using the statistical software R v.2.13.2 for Windows, except for CBR and NN Filtering results obtained exploiting the ANGEL tool [24].

2.4.1 Linear Regression

LR [14] is a statistical technique whereby a prediction model (Equation) is built, and represents the relationship between independent (e.g. number of Web pages) and dependent variables (e.g. total Effort).

The independent variables used with LR were selected beforehand via the Manual Stepwise Regression (MSWR) technique proposed by Kitchenham in [5], which enables the selection of only the independent variables that are significantly associated with the dependent variables, while also accounting for multicollinearity between variables [14]. The independent and dependent variables had to be transformed in order to comply with some of the assumptions of these techniques. The transformed variables were named the same way as their original variables, but preceded by Ln, which stands for 'natural log' transformation. Note that 1 was added to all the variables that contained zeroes, prior to being transformed. We also verified the stability of each model built using MSWR following the procedure suggested by Mendes [15].

2.4.2 Case-Based Reasoning

CBR is a branch of Artificial Intelligence where knowledge of similar past cases is used to solve new cases [24]. Herein cases

represent projects, which are characterized by features (variables), and stored in a case base; they are later used to find similar projects to the one chosen as target. When using CBR, one has to decide [24]: what relevant project features to use to characterize a project, the appropriate similarity function to obtain similar projects, the number of most similar projects to consider for estimation, and the analogy adaptation strategy for generating the estimation. Like [20][21][22], we used the Euclidean distance as similarity measure, all the numerical features, and the mean effort from the most similar, the two most similar, and the three most similar projects in the case base.

2.4.3 Nearest Neighbor Filtering

The NN Filtering technique [2] uses the k-Nearest Neighbor (k-NN) method to measure the similarity between projects in a validation and training sets by computing the Euclidean distance between those projects' features. The aim is to reduce the size of training sets to only include the most similar projects to those in the validation set. In order to apply the NN technique, for each project p in a validation set, we selected from its corresponding training set the k most similar projects to p (k=10[1]). Thus, for example, when using the CC datasets as training sets, we had the following: for each SC data set, with N observations, we obtained a total of 10 × N similar projects from the CC dataset. Since a project can be a nearest neighbor of many projects in the SC data set, these 10 × N observations may contain duplicates that were eliminated to form the final training set. To measure the similarity between projects we used all dataset features except for the effort data, since this corresponds to a real life situation, where the development effort is unknown when estimation takes place. Note that the NN Filtering must be used in combination with another technique in order to obtain effort estimates for projects. Therefore, within the context of this work, in order to address questions 3 and 4, filtering was employed prior to using MSWR+LR to obtain effort estimates.

Table 5 and 6 report the descriptive statistics for the datasets obtained by applying the NN Filtering to select the training sets representing CC data, when using as validation sets SC1 and SC2, respectively. In the following we refer to this datasets as NN_CC1 and NN_CC2. We can observe that standard deviation values are in general smaller with respect to the ones observed for CC1 and CC2 (see Tables 3 and 4) thus suggesting that filtering allowed us to obtain more homogenous datasets. Moreover, we can observe that in both NN_CC1 and NN_CC2 datasets the average number of employed languages remains the same observed for CC1 and CC2, while the average size of development team decreases and the average developers experience increases, achieving for both variables values more close to those observed for single-company projects (i.e., SC1 and SC2). Moreover, the average application size of cross-company projects contained in NN_CC1 is bigger with respect to the application size of cross-company projects contained in CC1, thus resulting also in an increment of the average effort which became more close to the average effort observed for single-company projects. On the contrary, the average application size of cross-company projects contained in NN_CC2 is smaller with respect to the application size of cross-company projects contained in CC2 thus resulting in a decrement of the average effort. However, this decrement is very low since the average number of high-effort features/functions increased and so the average effort spent for single-company projects was still higher than those spent for cross-company ones.

[1] Notice that we experimented NN Filtering using different values for k (i.e., 5, 10, 20) and k=10 provided the best results. This value was also employed in [2].

Table 5. Summary statistics for variables in NN_CC1

	Mean	Median	St.Dev.	Min	Max
nLang	3.21	3.00	1.56	1.00	6.00
DevTeam	2.25	2.50	1.29	1.00	6.00
TeamExp	4.83	5.00	2.23	1.00	10.00
TotWP	99.63	35.50	277.35	3.00	1390.00
NewWP	87.25	29.00	267.22	3.00	1333.00
TotImg	124.96	73.50	168.04	8.00	780.00
NewImg	52.46	18.50	118.52	0.00	583.00
Fots	1.17	1.00	1.58	0.00	6.00
HFotsA	1.00	0.50	1.29	0.00	4.00
Hnew	1.71	0.00	3.32	0.00	14.00
TotHigh	2.71	1.00	4.03	0.00	16.00
FotsA	2.92	2.00	3.15	0.00	11.00
New	3.25	2.00	4.07	0.00	13.00
TotNHigh	6.17	6.00	5.43	0.00	22.00
TotEff	454.99	85.00	1143.67	18.00	5000.00

Table 6. Summary statistics for variables in NN_CC2

	Mean	Median	St.Dev.	Min	Max
nLang	4.47	4.00	1.13	3.00	8.00
DevTeam	1.92	2.00	1.11	1.00	5.00
TeamExp	5.25	5.00	1.89	1.00	10.00
TotWP	47.86	20.00	65.73	3.00	300.00
NewWP	22.72	9.00	51.11	0.00	300.00
TotImg	87.33	19.00	215.76	0.00	1238.00
NewImg	14.69	1.00	23.63	0.00	87.00
Fots	2.86	1.50	3.26	0.00	10.00
HFotsA	47.17	1.00	127.60	0.00	611.00
Hnew	0.53	0.00	1.34	0.00	6.00
TotHigh	47.69	2.50	127.41	0.00	611.00
FotsA	2.25	1.00	3.71	0.00	20.00
New	5.42	3.50	5.63	0.00	19.00
TotNHigh	7.67	6.00	7.63	0.00	35.00
TotEff	482.89	89.00	1008.90	6.00	3644.00

2.5 Steps to Follow to Answer Our Research Questions

This Section details the steps that need to be carried out to answer each of the research questions this study investigated, by exploiting the data set, the modeling techniques, and the evaluation criteria described in the previous Sections. Note that to answer RQ1 and RQ2 we employed the same steps reported in [21].

Steps to follow to answer RQ1:

1) Apply MSWR+LR to build a cross-company cost model using the cross-company data set. Not applicable to CBR.

2) If model is not linear, transform the model back to the raw data scale. Not applicable to CBR.
3) Use the model in step 2 to estimate effort for each of the single-company projects. The single-company projects are the validation set used to obtain effort estimates (i.e., SC1 and SC2 datasets). The estimated effort obtained for each project is also used to calculate accuracy statistics (e.g. MRE). The equivalent for CBR is to use the cross-company data set as a case base to estimate effort for each of the single-company projects.
4) The overall model accuracy is aggregated from the validation set (e.g. MMRE, MdMRE), for both techniques.

These steps are used to simulate a situation where a company uses a cross-company data set to estimate effort for its new projects.

Steps to follow to answer RQ2:

1) Apply MSWR+LR to build a single-company cost model using the single-company data set (i.e., SC1 and SC2). Not applicable to CBR.
2) Obtain the prediction accuracy of estimates for the model obtained in 1) using a leave-one-out cross-validation. Cross-validation is the splitting of a data set into training and validation sets. Training sets are used to build models and validation sets are used to validate models. A leave-one-out cross-validation means that the original data set is divided into *n* different subsets (*n* is the size of the original data set) of training and validation sets, where each validation set has one project. The equivalent for CBR is to use the single-company dataset (i.e., SC1 and SC2) as a case base, after removing one project, and then to estimate effort for the project that has been removed. This step is iterated *n* times, each time removing a different project. The overall model accuracy is aggregated across the *n* validation sets. Same for CBR.
3) Compare the accuracy obtained in Step 3 to that obtained for the cross-company data set. Same for CBR.

Steps 1 to 3 simulate a situation where a company builds a model using its own data set and then uses this model to estimate effort for its new projects.

Steps to follow to answer RQ3:

1) Apply the filtering method described in Section 2.4.3 to the cross-company data (i.e., CC1 and CC2), using the SC1 and SC2 data as validation set, respectively, in order to obtain filtered cross-company data sets (i.e., NN_CC1 and NN_CC2).
2) Use NN_CC1 and NN_CC2 datasets as training sets, following the same steps provided for RQ1 with MSWR.

These steps are used to simulate a situation where a company uses a pre-filtered existing cross-company data set to estimate effort for its new projects.

Steps to follow to answer RQ4:

1) Compare the CC and SC models built following the steps provided for RQ1 and RQ2 with the one built using the NN_CC1 and NN_CC2 datasets.

Finally, we employed the mean and median-based predictions (i.e., effort estimate is respectively the mean or median effort for the training set) as benchmarks for our prediction models since performing similarly or worse than these benchmarks is a strong indicator of poor performance and this analysis can contribute to answer RQ1 and RQ3.

3. MODELS CONSTRUCTION

3.1 Building Cross-Company Models

3.1.1 S5CCM1

The CC model S5CCM1 built using CC1 dataset formed by the Tukutuku dataset after excluding the 31 projects of one of the single-companies (i.e., SC1 dataset) has an adjusted R^2 of 0.70, thus explaining 70% of the variation in effort (see Table 7).

Table 7. Initial S5CCM1 Model

Independent Variables	Coeff.	Std. Error	t	p>|t|
(Intercept)	2.431	0.151	16.102	< 2e-16
LnNewWP	0.436	0.055	7.858	5.37e-13
LnNewImg	0.199	0.045	4.421	1.81e-05
LnHNew	1.029	0.104	9.934	< 2e-16

To identify influential data points we calculated the Cook's distance values [2] for all 164 projects. Results revealed that one project has distance higher than 0.073, thus it was immediately removed from the data analysis [14], while other seven projects have distances higher than 0.024 but smaller than 0.073. These were removed in order to test the model stability, by observing the effect of their removal on the model. Since the model coefficients remained stable and its goodness of fit improved (i.e., the adjusted R^2 improved from 0.71 to 0.76), the highly influential projects were retained in the data analysis. The revisited baseline model is presented in Table 8[2].

The Equation as read from the final model's output is:

$$LnTotEffort = 2.406 + 0.459 \times LnNewWP + 0.172 \times LnNewImg + 1.925 \times LnHNew \quad (2)$$

which, when converted back to the raw data scale, gives the Equation:

$$TotEffort = 11.090 \times LnNewWP^{0.456} \times nLnNewImg^{0.172} \times LnHNew^{1.025} \quad (3)$$

Table 8. Final S5CCM1 Model

Independent Variables	Coeff.	Std. Error	t	p>|t|
(Intercept)	2.406	0.149	16.203	< 2e-16
LnNewWP	0.459	0.055	8.332	3.5e-14
LnNewImg	0.172	0.045	3.806	0.000201
LnHNew	1.025	0.102	10.078	<2e-16

3.1.2 S5CCM2

The CC model S5CCM2 was built using CC2 dataset formed by the entire Tukutuku dataset, after excluding the 18 projects from the second single-company being used herein (SC2). It presented an adjusted R^2 of 0.74, thus explaining 74% of the variation in effort (see Table 9). The Cook's distance values calculated for each of the 177 projects revealed that four projects have distances higher than 0.069, thus they were immediately removed from the data analysis [14]. Another seven projects had distances higher than 0.023 but smaller than 0.069, thus they were removed in

[2] The residual and Q-Q plots for each of the models built in this section were omitted due to shortage of space.

order to test the model stability. Since the model coefficients remained stable and the adjusted R^2 improved from 0.78 to 0.80, the influential projects were retained in the data analysis. The revisited baseline model is presented in Table 10.

The Equation as read from the final model's output is:

$$LnTotEffort=0.443+0.612\times LnTotWP+0.729\times LnLang+0.681\times LnDevTeam + 0.803\times LnTotNHigh - 0.358\times LnHFotsA \quad (4)$$

which, when converted back to the raw data scale, gives the Equation:

$$TotEffort = 1.542 \times TotWP^{0.612} \times nLang^{0.729} \times DevTeam^{0.681} \times TotNHigh^{0.803} \times LnHFotsA^{(-0.358)} \quad (5)$$

Table 9. Initial S5CCM2 Model

Independent Variables	Coeff.	Std. Error	t	p>|t|
(Intercept)	0.713	0.313	2.280	0.023862
LnLang	0.649	0.190	3.428	0.000763
LnDevTeam	0.746	0.142	5.264	4.19e-07
LnTotWP	0.543	0.066	8.271	3.59e-14
LnTotHigh	0.792	0.117	6.754	2.16e-10
LnHFotsA	-0.324	0.117	-2.775	0.006139

Table 10. Final S5CCM2 Model

Independent Variables	Coeff.	Std. Error	t	p>|t|
(Intercept)	0.443	0.305	1.452	0.148432
LnLang	0.729	0.185	3.943	0.000118
LnDevTeam	0.681	0.153	4.462	1.49e-05
LnTotWP	0. 612	0.065	9.420	< 2e-16
LnTotHigh	0.803	0.112	7.159	2.46e-11
LnHFotsA	-0.358	0.113	-3.174	0.002

3.2 Building Cross-Company Models Using NN Filtered Data

3.2.1 S5CCM3

The model S5CCM3 was built using the NN_CC1 data (i.e., the ones obtained by applying NN Filtering on CC1) and presented an adjusted R^2 of 0.70 thus explaining 70% of the variation in effort (see Table 11).

Table 11. S5CCM3 model

Independent Variables	Coeff.	Std. Error	t	p>|t|
(Intercept)	2.434	0.560	4.347	0.000283
LnHNew	0.881	0.292	3.018	0.006552
LnTotWP	0.523	0.175	2.993	0.006934

The results from Cook's distance revealed that all projects have distances less than 0.48, while two projects have distances higher than 0.16 thus these two projects were removed in order to test the model stability, by observing the effect of their removal on the model. Since the model coefficients remained stable and the adjusted R^2 improved from 0.70 to 0.76, the highly influential projects were retained in the data analysis and the final model is the same one presented in Table 11. The Equation as read from the final model's output is:

$$LnTotEffort = 2.434 + 0.881\times LnHNew + 0.523\times LnTotWP \quad (6)$$

which, when converted back to the raw data scale, gives the Equation:

$$TotEffort = 11.404 \times LnHNew^{0.881} \times LnTotWP^{0,523} \quad (7)$$

3.2.2 S5CCM4

The model S5CCM4 was built using the NN_CC2 data (i.e., data obtained by applying NN Filtering to CC2); it presented an adjusted R^2 of 0.69, thus explaining 69% of the variation in effort (see Table 12).

The Cook's distance values revealed that all projects have distances less than 0.33 while two projects have distances higher than 0.11, thus only these two projects were removed in order to test the model stability, by observing the effect of their removal on the model. Since the model coefficients remained stable and the adjusted R^2 improved from 0.69 to 0.75, the highly influential projects were retained in the data analysis and the final model is the one presented in Table 12. The Equation as read from the final model's output is:

$$LnTotEffort = -2.186 + 2.650\times LnLang + 0.950\times LnTotWP \quad (8)$$

which, when converted back to the raw data scale, gives the Equation:

$$TotEffort = 0.112 \times LnLang^{2.650} \times LnTotWP^{0.950} \quad (9)$$

Table 12. S5CCM4 Model

Independent Variables	Coeff.	Std. Error	t	p>|t|
(Intercept)	-2.186	1.005	-2.176	0.0369
LnLang	2.650	0.639	4.150	0.000
LnTotWP	0.950	0.126	7.514	1.22e-08

3.3 Building Single-Company Models

3.3.1 S5SCM1

The model built using the SC1 dataset (31 projects) presented an adjusted R^2 of 0.94, thus explaining 94% of the variation in effort (see Table 13).

Table 13. Best S5SCM1 Model

Independent Variables	Coeff.	Std. Error	t	p>|t|
(Intercept)	2.092	0.153	13.673	6.46e-14
LnHFots	0.920	0.046	19.868	< 2e-16
LnNewImg	0.234	0.071	3.323	0.00249

Cook's distance values showed that all projects had distances lower than 0.39; five projects had distances higher than 0.13, thus they were removed in order to test the model stability. Since the model coefficients remained stable and the adjusted R2 (goodness of fit) improved from 0.94 to 0.96, the highly influential projects were retained in the data analysis. So the final model is the one presented in Table 13 and the Equation as read from the final model's output is:

$$LnTotEffort = 2.092 + 0.920\times LnHFots + 0.234\times LnNewImg \quad (10)$$

which, when converted back to the raw data scale, gives the Equation:

$$TotEffort = 8.101 \times HFots^{0.920} \times NewImg^{0.234} \quad (11)$$

3.3.2 S5SCM2

The model built using the SC2 dataset presented an adjusted R^2 of 0.70, thus explaining 70% of the variation in effort (see Table 14).

Table 14. Best S5SCM2 Model

Independent Variables	Coeff.	Std. Error	T	p>|t|
(Intercept)	2.600	0.365	7.117	2.44e-06
LnTotWP	0.694	0.108	6.410	8.63e-06

Cook's distance values showed that no projects had distances higher than 0.66 or higher than 0.23, thus no observations were removed and the final model is the one presented in Table 14. The Equation as read from the final model's output is:

$$LnTotEffort = 2.6 + 0.694 \times LnTotWP \quad (12)$$

which, when converted back to the raw data scale, gives the Equation:

$$TotEffort = 13.464 \times TotWP^{0.694} \quad (13)$$

4. RESULTS

4.1 Addressing RQ1

RQ1: How successful is a cross-company dataset at estimating effort for projects from a single company?

As shown in Table 15, the prediction accuracy obtained by applying the CC models S5CCM1 and S5CCM2 to respectively the validation sets SC1 and SC2, is quite poor. We also report the accuracy obtained using the mean and median effort. The prediction accuracy of both CC models is superior to estimates obtained using the mean effort (does not corroborate S3 results); however their accuracy is worse than that obtained using the median effort (corroborates S3 results). These results were statistically significant both on SC1 (p-values=0.04 and 0.02 with small and medium effect size, respectively) and SC2 (p-values=0.00 with medium and high effect size), as revealed by the Wilcoxon test.

Table 15 also reports the prediction accuracy obtained employing CBR to the validation sets SC1 and SC2, respectively; notice that due to space constraints only the best results are reported herein and the number of analogies used to obtain each of these best results is denoted as CBRk, where k is the number of the employed analogies. The obtained results highlighted that on the SC1 validation set CBR provided superior estimates with respect to those obtained using the mean effort and comparable with respect to the ones obtained using the median effort (different results to the one in S3). These results are confirmed by the Wilcoxon tests highlighting that CBR results were significantly better than those achieved by mean effort (p-value=0.00), while no significant difference (p-value=0.56) has been found with respect to those achieved by median effort (small effect sizes). As for the SC2 validation set, CBR provided significantly worse estimates than those achieved by using median effort (p-value=0.00), while there was no significant difference with respect to the ones obtained using mean effort (p-value=0.91) (in both cases we noticed small effect sizes).

4.2 Addressing RQ2

RQ2: How successful is the use of a cross-company dataset, compared to a single-company dataset, for effort estimation?

Table 16 reports the prediction accuracy obtained by applying a leave-one-out cross-validation procedure to the two single-company sets SC1 and SC2, using respectively the SC models described in Section 3.3 (i.e., S5SCM1 and S5SCM2). To address our second research question we used the Wilcoxon test to compare: i) the absolute residuals for the 31 SC1 projects with the single-company model (S5SCM1) to those obtained using 31 SC1 projects with the S5CCM1 model; and i) the absolute residuals for the 18 SC2 projects with the single-company model (S5SCM2) to those obtained using 18 SC2 projects with the S5CCM2 model. Both models S5SCM1 and S5SCM2 presented statistically significantly superior accuracy than the S5CCM1 and S5CCM2 models (p-values=0.00) with medium and high effect size, respectively. Both results corroborate those from S3. We have also compared the accuracy of the single-company models to the mean- and median-based effort models (see Table 16). The S5SCM1 model presented significantly better accuracy than both mean and median effort (p-values=0.00), with high and small effect size, respectively. On the contrary, the S5SCM2 model provided comparable results with respect to these benchmarks (i.e., not significant difference was found – p-values= 0.13 and 0.18 – with small effect). Both results corroborate those from S3.

Table 15. Prediction accuracy statistics for CC Models

Val. Set	Estimates based on	MAR	MMRE	MdMRE	Pred(25)
SC1	S5CCM1	584.91	0.81	0.61	0.06
	CBR1	510.58	1.38	0.82	0.23
	Mean	700.90	5.37	3.51	0.00
	Median	549.69	0.87	0.70	0.23
SC2	S5CCM2	133.60	0.58	0.61	0.13
	CBR1	546.47	2.35	0.80	0.11
	Mean	336.08	4.02	2.13	0.06
	Median	97.50	0.57	0.56	0.22

Table 16. Prediction accuracy statistics for SC Models

Val. Set	Estimates based on	MAR	MMRE	MdMRE	Pred(25)
SC1	S5SCM1	252.25	0.33	0.23	0.52
	CBR3	262.04	0.61	0.36	0.39
	Mean	826.64	7.81	5.34	0.00
	Median	564.19	1.31	0.67	0.03
SC2	S5SCM2	62.27	0.37	0.27	0.50
	CBR3	68.43	0.44	0.44	0.33
	Mean	85.00	0.94	0.50	0.39
	Median	85.53	1.00	0.48	0.39

Table 16 also reports the prediction accuracy obtained by applying a leave-one-out cross-validation procedure to the two single-company sets SC1 and SC2, using CBR. To address RQ2 we compared: i) the absolute residuals for the 31 SC1 projects with the single-company model (S5SCM1) to those obtained using 31 SC1 projects with CBR; and ii) the absolute residuals for the 18 SC2 projects with the single-company model (S5SCM2) to those obtained using 18 SC2 projects with CBR.

Similarly to the regression-based models, the results provided by CBR were significantly worse than those obtained by using CBR with SC data (p-values=0.00), however the correspondent effect sizes were small. Both results corroborate those from S3. We have also compared the accuracy of CBR on single-company sets to the mean- and median-based effort models (see Table 17). CBR1 provided significantly better results than using mean and median effort (p-values=0.00) with small and medium effect size, respectively, on SC1, while there were no significant differences

with mean and median (p-values= 0.16 and 0.23, respectively) on SC2 with a small effect size. These results corroborate those from S3.

4.3 Addressing RQ3

RQ3: How successful is a filtered cross-company dataset at estimating effort for projects from a single company?

Table 17 shows the prediction accuracy obtained by applying the CC models built using filtered cross-company dataset (i.e., S5CCM3 and S5CCM4) to the validation sets SC1 and SC2, respectively. We also report the accuracy obtained using the mean and median effort. Both S5CCM3 and S5CCM4 provided better accuracy measures than the mean-based model and comparable to the median effort model. The results for S5CCM3 were corroborated using the Wilcoxon test (p-values=0.00), with a medium effect size; while the significance test showed no statistical difference between S5CCM4 and both the mean-(p-value= 0.13) and median-based (p-value=0.18) models with high and medium effect size, respectively.

4.4 Addressing RQ4

RQ4: How successful is the use of a filtered cross-company dataset, compared to (a) the non-filtered cross-company dataset, and (b) a single-company dataset?

To address the fourth research question we used the Wilcoxon test to compare: i) the absolute residuals for the 31 SC1 projects with the cross-company model (S5CCM1) to those obtained using 31 SC1 projects with the S5CCM3 model; ii) the absolute residuals for the 18 SC2 projects with the cross-company model (S5CCM2) to those obtained using 18 SC2 projects with the S5CCM4 model; iii) the absolute residuals for the 31 SC1 projects with the single-company model (S5SCM1) to those obtained using 31 SC1 projects with the S5CCM3 model; and iv) the absolute residuals for the 18 SC2 projects with the single-company model (S5SCM2) to those obtained using 18 SC2 projects with the S5CCM4 model. The Wilcoxon test showed that both models S5CCM3 and S5CCM4 presented statistically significantly superior accuracy than the S5CCM1 and S5CCM2 models (p-values=0.00), respectively, with high effect size. Thus, revealing that the use of a filtered cross-company dataset allowed us to achieved better results with respect to the non-filtered cross-company dataset. As for the comparison with the single-company models, we can observe that the S5CCM3 model is significantly worse than S5SCM1 (p-value=0.00) with a small effect size. On the other hand, no statistically significant difference has been found between S5CCM3 and S5SCM2 models (p-value=0.14) with a small effect size.

The above results suggest us that the use of a filtered cross-company datasets is successful with respect to the use of non-filtered cross-company datasets and comparable with single-company datasets.

Table 17. Prediction accuracy statistics for NN_CC Models

Val Set	Estimates based on	MAR	MMRE	MdMRE	Pred(25)
SC1	S5CCM3	56.24	0.58	0.61	0.13
	Mean	708.6	5.54	3.64	0.00
	Median	549.6	0.87	0.70	0.23
SC2	S5CCM4	68.55	0.37	0.35	0.33
	Mean	319.8	3.86	2.03	0.06
	Median	92.94	0.58	0.51	0.17

5. THREATS TO VALIDITY

There are several factors that can bias the validity of empirical studies. Here we consider three types of validity threats: construct validity, related to the agreement between a theoretical concept and a specific measuring device or procedure; conclusion validity, related to the ability to draw statistically correct conclusions; external validity, related to the ability to generalise the achieved results. As highlighted by Kitchenham *et al.* [6], to satisfy construct validity a study has "to establish correct operational measures for the concepts being studied". This means that the study should represent to what extent the predictor and response variables precisely measure the concepts they claim to measure. Thus, the choice of the features and how to collect them represent the crucial aspects. As discussed in Section 2 the size measures and cost drivers used in the Tukutuku database, and therefore in our study, have been obtained from the results of a survey investigation and have also been confirmed by an established Web company and a second survey [17]. Consequently, it is our belief that the variables identified are measures that are meaningful to Web companies and are constructed from information their customers can provide at a very early stage in the project development. As for data quality, it was found that at least for 93.8% of Web projects in the Tukutuku database effort values were based on more than just guesstimates [17]. With respect to the conclusion validity we carefully applied the statistical tests, verifying all the required assumptions. Moreover, we also employed effect size to assess the relevance of the obtained results. As for external validity, let us observe that the Tukutuku dataset comprises data on projects volunteered by individual companies, and therefore it does not represent a random sample of projects from a defined population. This means that we cannot conclude that the results of this study promptly apply to other companies different from the ones that volunteered the data used here. However, we believe that Web companies that develop projects with similar characteristics to those used in this paper may be able to apply our results to their Web projects.

6. RELATED WORK USING FILTERING[3]

To the best of our knowledge, all the previous studies that investigated the use of filtering (or clustering) techniques as pre-processor for comparing CC vs. SC data in the context of effort estimation did so using datasets from the PROMISE repository, which represent software (not Web) project data, and also investigated analogy based techniques as prediction method while we combined the use of filtering with MSWR. In particular, motivated by the fact that the use of filtering techniques improved the use of CC data in the context of defect prediction [2], Kocaguneli et al. [10] slightly modified an analogy based effort predictor (i.e., ABE0) to include instance selection as a pre-processor for a study on cross vs. single resource effort estimation. In particular, their relevancy filter is similar to the NN filtering used by Turhan et.al. [10], except that there is no need to pre-specify the number of analogies k to be used for estimation. Thus, their estimation technique is a small variant of ABE0 which works in two passes: (i) remove training instances implicated in poor decisions, (ii) select those instances nearest the test instance. In the empirical study carried out with three datasets contained in the PROMISE repository, they found that after instance selection,

[3] The Related work comparing CC vs. SC for Web effort prediction has been reported in the Introduction Section.

the performance differences in the predictors learned from cross or within data were statistically insignificant. This work was extended in [11] where ABE0 and another ABE variant, namely TEAK, were assessed using other three datasets from PROMISE repository. In particular, TEAK is a variance-based instance selector that discards training data associated with regions of high estimation variance augmenting ABE0 both with instance selection and an indexing scheme for filtering relevant training examples. The obtained results showed that instance selection on cross sources improved analogy based estimators performance to an extent where it is no worse than within data confirming previous conclusions [10] in a much larger scale with 4 error measures and 21 different cases. Starting from the results of those studies, Menzies et al. [23] investigated the data heterogeneity and its influence on conclusion instability for effort and defect prediction by analyzing the use of both clustering and inferred dimensions as techniques for localized reasoning.

7. CONCLUSIONS

In this study, we have analyzed the accuracy of Web effort estimates obtained using a cross-company and single-company data sets, and results are as follows:

- The cross-company data set provided poor predictions for the single-company projects and much worse predictions than the single-company data set when employing either MSWR+LR or CBR. These results corroborate those obtained in S3 using a larger dataset and it is interesting to observe that larger amounts of data do not significantly affect the trends seen in previous analysis. We can argue that the poor performance of CC models could be due to the heterogeneity of CC data.

- The use of NN Filtering improved the prediction accuracy of cross-company models when estimating single-company projects, indeed the obtained results were significantly better than those obtained for cross-company models without any filtering and comparable to those using single-company models. This suggests that filtering techniques, creating more homogenous training set may provide the means to improve the effectiveness of CC models. This results corroborate those reported in the literature on traditional software projects for software effort estimation and defect prediction. Further research in this area is warranted also investigating other filtering mechanisms such as the ones employed in [12][13] to improve analogy based effort estimation.

8. ACKNOWLEDGMENTS

We would like to thank all those companies that have volunteered data on their projects to the Tukutuku database.

9. REFERENCES

[1] Conte, S. D., Dunsmore, H. E., Shen, V. Y.,1986. Software Engineering Metrics and Models. Benjamin-Cummins.

[2] Turhan, B., Menzies, T., Bener, A., Di Stefano,J., 2009. On the relative value of cross-company and within-company data for defect prediction. ESE,14:540-578.

[3] Cohen, J., 1988. Statistical power analysis for the behavioral sciences. Lawrence Earlbaum Ass., 2nd edition.

[4] Cook, R.D., 1977. Detection of influential observations in linear regression. Technometrics, 19, 15-18.

[5] Kitchenham B., 1998. A Procedure for Analyzing Unbalanced Datasets. IEEE TSE, 278-301.

[6] Kitchenham, B.A., Pickard, L., Peeger, S. 1995. Case studies for method and tool evaluation. IEEE Software 12(4): 52-62.

[7] Kitchenham, B., Pickard, L. M., MacDonell, S. G., Shepperd, M. J, 2001. What accuracy statistics really measure. IEE Proceedings Software 148 (3) 81–85.

[8] Kitchenham,B., Mendes, E., 2004. A Comparison of Cross-company and Single-company Effort Estimation Models for Web Applications, in Procs. of EASE, 47-55.

[9] Kitchenham B.A., Mendes, M., Travassos, G.H., 2007. Cross versus Within-Company Cost Estimation Studies: A Systematic Review. IEEE TSE 33(5):316-329.

[10] Kocaguneli, E., Gay, G., Menzies, T., Yang, Y., Keung, J. W., 2010. When to use data from other projects for effort estimation. In Procs. of ASE, 321-324

[11] Kocaguneli, E., Menzies, T., 2011. How to Find Relevant Data for Effort Estimation? In Procs of ESEM, 255-264.

[12] Kocaguneli,E., Menzies,T., Bener,A.B., Keung,J.W., 2012. Exploiting the Essential Assumptions of Analogy-Based Effort Estimation. IEEE TSE 425-438.

[13] Li, Y.F., Xie, M., Goh, T.N., 2009. A Study of Project Selection and Feature Weighting for Analogy based Software Cost Estimation. JSS, 82(2), 241–252.

[14] Maxwell, K., 2002. Applied Statistics for Software Managers. Software Quality Institute Series, Prentice Hall.

[15] Mendes, E., 2008. Web Cost Estimation and Productivity Benchmarking. ISSSE: 194-222.

[16] Mendes,E., Mosley, N., Counsell, S., 2002. Comparison of Length, complexity and functionality as size measures for predicting Web design and authoring effort, IEE Procs. Software 149 (3), 86-92.

[17] Mendes, E., Mosley, N., Counsell, S., 2003. Investigating early web size measures for web cost estimation. In Procs. Evaluation and Assessment in Software Engineering, 1–22.

[18] Mendes, E., Mosley, N. and Counsell, S., 2005. The Need for Web Engineering: An Introduction. Web Engineering, Springer-Verlag, Mendes, E. and Mosley, N. (Eds.)

[19] Mendes, E., Mosley, N., Counsell, S. 2005. Investigating Web Size Metrics for Early Web Cost Estimation. JSS, 77(2), 157-172.

[20] Mendes, E., Kitchenham B.A., 2004. Further Comparison of Cross-Company and Within Company Effort Estimation Models for Web Applications. In Procs of METRICS, IEEE Computer Society, 348-357.

[21] Mendes, E., Di Martino, S., Ferrucci, F., Gravino, C., 2007. Effort Estimation: How Valuable is it for a Web company to Use a Cross-company Data Set, Compared to Using Its Own Single-company Data Set?. In Procs of WWW07, 963-972.

[22] Mendes, E., Di Martino, S., Ferrucci, F., Gravino, C., 2008. Cross-company vs. single-company web effort models using the Tukutuku database: An extended study. JSS 81, 673–690.

[23] Menzies, T.. Butcher, A., Marcus, A., Zimmermann, T., Cok, D. R., 2011. Local vs. global models for effort estimation and defect prediction. In Procs. of ASE, 343-351.

[24] Shepperd, M.J., and G. Kadoda, 2001. Using Simulation to Evaluate Prediction Techniques, in Proceedings IEEE 7th Intl Software Metrics Symposium, London, UK, 349-358.