Proceedings of the Federated Conference on Computer Science and Information Systems pp. 763–770
DOI: 10.15439/2019F222
ISSN 2300-5963 ACSIS, Vol. 18

# Handling of Categorical Data in Software Development Effort Estimation: A Systematic Mapping Study

Fatima Azzahra Amazal
LabSIV, Department of Computer Science, Faculty of Science, Ibn Zohr University, BP 8106, 80000 Agadir, Morocco
Email: amazal.fatimaazzahra@gmail.com

Ali Idri
Software Projects Management Research Team, ENSIAS, Mohamed V University, Madinate Al Irfane, 10100 Rabat, Morocco
Email: idri@ensias.ma

***Abstract*—Producing reliable and accurate estimates of software effort remains a difficult task in software project management, especially at the early stages of the software life cycle where the information available is more categorical than numerical. In this paper, we conducted a systematic mapping study of papers dealing with categorical data in software development effort estimation. In total, 27 papers were identified from 1997 to January 2019. The selected studies were analyzed and classified according to eight criteria: publication channels, year of publication, research approach, contribution type, SDEE technique, Technique used to handle categorical data, types of categorical data and datasets used. The results showed that most of the selected papers investigate the use of both nominal and ordinal data. Furthermore, Euclidean distance, fuzzy logic, and fuzzy clustering techniques were the most used techniques to handle categorical data using analogy. Using regression, most papers employed ANOVA and combination of categories.**

## I. Introduction

THE competitiveness of software companies relies on the successful management of their software projects. One of the most important and difficult tasks in software project management is how to accurately estimate the effort needed to develop a software product. This task is known as software development effort estimation (SDEE). Delivering reliable and accurate estimates remains a challenging objective for software companies due to several factors including the human factor, the variety of software projects, the inherent uncertainty of feature measurement, and the diversity of development environments [1]. In attempt to get accurate predictions, various SDEE techniques have been proposed. These techniques fall into three main types [2]: parametric models [3], [4], machine learning (ML) models [5]-[10] and expert judgment [11].

SDEE techniques build their predictions based on a set of attributes (also called features or cost drivers) that characterize software projects [12], [13]. Most of these techniques derive their predictions based on numerical attributes. However, the information available at the early stages of the software life cycle is more categorical than numerical. Furthermore, the datasets used to build and validate SDEE models involve a high number of categorical data. For example, in COCOMO'81 dataset [14], 15 attributes out of 17 are measured on a scale composed of six categories: very low, low, nominal, high, very high, and extra high. Another example is the International Software Benchmarking Standards Group (ISBSG) dataset [15], in which numerous attributes such as programming language, application type and development platform are measured on a nominal scale.

Categorical attributes may be measured on a nominal or ordinal scale. The nominal scale type allows the classification of entities into different categories [16], for example, primary programming language may be classified into five categories: Visual basic, C, Cobol, Visual C++, Oracle. Unlike the nominal scale type in which there is no order between the categories of entities, the ordinal scale type enables ranking the categories in a specific order [16]. An example of ordinal attributes is the application experience which may be measured as: ‘low’, ‘nominal’, ‘high’, and ‘very high’. To deal with this kind of attributes, different approaches were used in SDEE literature [17]-[21].

In this paper, a Systematic Mapping Study (SMS) is performed to investigate the use of categorical data to estimate software development effort. As pointed out in [22], a systematic map is a method that concentrates on building a classification scheme and categorizing primary research studies in a specific domain with respect to a set of defined categories. Thus, it provides a common starting point for many researchers [23]. To the best of the authors' knowledge, no systematic mapping study has been carried out with focus on how to handle categorical data in SDEE.

This SMS aims to: 1) identify the existing SDEE papers dealing with categorical data and published from 1997 to January 2019; and 2) analyze and classify the selected papers according to 8 criteria: publication channels, year of publication, research approach, contribution type, SDEE technique, Technique used to handle categorical data, types of categorical data and datasets used.

This paper is structured as follows: Section II presents the research methodology adopted to carry out this SMS.

Section III, reports the results of the mapping study. Section IV presents the implications for research and practice. Conclusions and future work are presented in Section V.

## II. RESEARCH METHODOLOGY

In this study, the systematic mapping process suggested by Kitchenham and Charters [24] is used. According to Kitchenham, a mapping study aims to identify the research trends related to a specific topic and classify research works with respect to a set of defined criteria [22], [24]. The mapping process used comprises the following five steps: (1) define the mapping questions, (2) conduct an exhaustive search for candidate papers, (3) select studies, (4) extract data, and (5) summarize data. Each of these steps is described next.

### A. Mapping questions

Eight mapping questions (MQs) were formulated in this mapping study. Table I shows the MQs as well as their main motivations.

### B. Search Strategy

The aim of this step is to find the relevant SDEE papers that address the MQs listed in table I. To perform the search, four electronic databases were used: ACM Digital library, IEEE Xplore, Science Direct and Google Scholar. These libraries were chosen since they were used in previous systematic maps and reviews in SDEE to conduct the search for candidate papers [5], [25], [26]. All searches were restricted to the studies published between 1997 and January 2019.

TABLE I. MAPPING QUESTIONS

| ID | Mapping Question | Motivation |
|---|---|---|
| **MQ1** | Which publication sources are the main targets for SDEE papers dealing with categorical data? | To identify the main sources where SDEE studies with focus on categorical data can be found. |
| **MQ2** | How has the frequency of handling categorical data in SDEE papers changed over time? | To investigate the publication trends of SDEE studies dealing with categorical data over time. |
| **MQ3** | What are the research approaches of the selected papers? | To discover the research approaches used by SDEE studies with focus on categorical data. |
| **MQ4** | What are the contribution types of the selected papers? | To explore the contribution types of SDEE papers dealing with categorical data. |
| **MQ5** | Which technique investigates the most the use of categorical data in SDEE? | To identify the SDEE techniques that handle the most categorical data. |
| **MQ6** | How categorical data are handled in SDEE? | To determine the different ways of handling categorical data in SDEE. |
| **MQ7** | What are the most investigated types of categorical data in SDEE? | To identify the types of categorical data that are the most investigated in SDEE. |
| **MQ8** | What are the datasets used for validation? | To explore the datasets used in the selected papers as well as the Percentage of categorical features used in the experiments. |

To carry out the search using the four databases, a search string was defined. To do so, we derived the main terms based on the MQs. Then, we identified all alternative spellings and synonyms of the major terms. The Boolean operators OR and AND were used to combine the main terms [25], [26]. The final search string was formulated as follows:

(software OR system OR application OR product OR project OR development) AND (effort OR cost) AND (estimat* OR predict* OR assess*) AND (categorical OR nominal OR ordinal OR "non-quantitative") AND (feature OR attribute OR data OR "cost driver").

To ensure that no relevant paper was missed, we adopted a search process of two stages. In the first stage, we performed the search in the four electronic databases using the above search string to identify the set of candidate papers. In the second stage, we applied the inclusion and exclusion criteria on each of the candidate papers based on title, abstract, and keywords to decide on its relevance to our study. If necessary, the full paper was examined. The reference list of each of the relevant papers was scanned to check whether a SDEE study with focus on categorical data was leaved out in the first stage.

### C. Study Selection

The purpose of this step was to select the papers that are relevant to our SMS (i.e., papers that addressed the MQs). To achieve this, a set of inclusion and exclusion criteria were applied on each of the candidate papers by each of the authors of this study to decide whether it should be retained or discarded.

Inclusion criteria

- ✓ Studies with focus on how to handle categorical data to estimate software effort
- ✓ Studies in which a technique is proposed or extended and which enables software effort estimation using categorical data or a mixture of numerical and categorical data
- ✓ Studies comparing different techniques that handle categorical data

Exclusion criteria:

- ✓ SDEE studies in which categorical features are not handled or discarded
- ✓ SDEE studies for which the main objective is not deal with categorical data and which use only transformation to dummy variables
- ✓ SDEE studies that fuzzify numerical inputs to get linguistic values without dealing with categorical inputs
- ✓ SDEE studies with focus on missing categorical data
- ✓ Duplicate publications of the same paper (In this case, only the most complete study is included)
- ✓ Studies estimating maintenance or testing effort

Using the above criteria, the two researchers independently evaluate the candidate papers. Based on the title and abstract (if necessary full text), a researcher might categorize a candidate paper as "include", "Exclude", or "Uncertain". A paper that was categorized as "Include" ("Exclude") by both researchers was retained (discarded); otherwise, the paper was discussed until an agreement was reached.

### *D. Data Extraction Strategy and Synthesis Method*

Each of the selected papers was examined by both authors to extract the data necessary to answer the mapping questions of table I. To this end, a data extraction form was used and completed by both authors for each selected paper. Table II shows the data extraction form used in our mapping study.

The extracted data were, then, synthesized and summarized with respect to each MQ. To achieve this, a narrative synthesis approach was used. We also used some visualization charts such as pie charts and bubble plots to improve the presentation of the results obtained and facilitate their interpretation.

TABLE II. DATA EXTRACTION FORM

| |
|---|
| **Data extractor** |
| **Paper identifier** |
| **Author(s) name(s)** |
| **Article title** |
| (MQ1) Publication Channel |
| (MQ2) Publication year |
| (MQ3) Research approach (History-based evaluation, solution proposal, case study, theory, review, survey, other) |
| (MQ4) Contribution type (Technique, tool, comparison, validation, metric, model, framework) |
| (MQ5) SDEE Techniques used in the paper |
| (MQ6) Technique used to handle categorical data |
| (MQ7) Types of categorical data used in the study |
| (MQ8) Datasets used |

## III. RESULTS AND DISCUSSION

This section presents and discusses the results of our systematic mapping related to the questions of table I.

### *A. Overview of the selected studies*

The results of the selection process are shown in Fig. 1. As can be seen, 1226 candidate papers were retrieved by applying the search string described previously on the four electronic databases. Afterward, the inclusion and exclusion criteria were used to evaluate each of the candidate papers and decide whether it should be retained or discarded. The evaluation was based on the title, abstract, keywords, and full text of the candidate papers. This process resulted in 27 relevant papers. No additional relevant studies were identified by checking the reference lists of the selected studies.

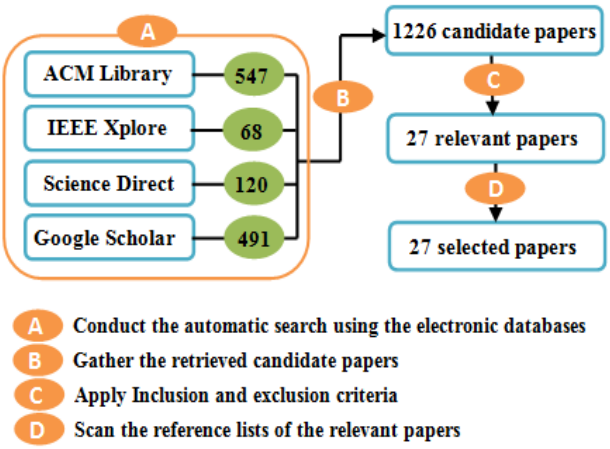


Fig. 1 Results of selection process

### *B. Publications Channels (MQ1)*

We identified two main publication channels in which the selected studies were published: journals and conferences. Specifically, among the 27 selected papers, 15 (55.56%) papers appeared in journals and 12 (44.44%) papers were presented at conferences. Tables III and IV shows the publication sources of the papers identified in journals and conferences respectively. The number of studies per publication source is given in the second column of each table. Three journals were identified with 2 or more papers dealing with categorical data in SDEE: Empirical Software Engineering, Information Software Technology, and IEEE Transactions on Software Engineering. Only one conference was identified with 2 papers: International Conference on Predictive Models in Software Engineering (PROMISE). The remaining sources (journals and conferences) were used once to publish SDEE studies with focus on categorical data.

TABLE III. PUBLICATION SOURCES OF JOURNAL PAPERS

| Publication venue | # of studies |
|---|---|
| Empirical Software Engineering | 4 |
| Information and Software Technology | 3 |
| IEEE Transactions on Software Engineering | 2 |
| The Journal of Systems and Software | 1 |
| International Journal of Intelligent Systems | 1 |
| International Journal of Computer Science and Engineering Survey | 1 |
| Software Quality Journal | 1 |
| Journal of Information Science and Engineering | 1 |
| IEEE Access | 1 |

TABLE IV. PUBLICATION SOURCES OF CONFERENCE PAPERS

| Publication venue | # of studies |
|---|---|
| International Conference on Predictive Models in Software Engineering | 2 |
| International Conference on Software Engineering Research, Management and Applications | 1 |
| Asia-Pacific Software Engineering Conference | 1 |

| | |
|---|---|
| Software Metrics Symposium | 1 |
| International Conference on Computer and Information Technology | 1 |
| International Conference on Software Engineering | 1 |
| International Conference on Computer Science and Automation Engineering | 1 |
| International Symposium on Software Metrics | 1 |
| International Conference on Enterprise Information Systems | 1 |
| International Conference on Communications, Circuits and Systems and West Sino Expositions | 1 |
| Empirical Software Engineering and Measurement | 1 |

### C. Publications Trends (MQ2)

To get a global picture of the publication trends of SDEE papers dealing with categorical data, we analyzed the distribution of the selected studies over time. Fig. 2 shows the number of papers per year from 1997 to January 2019. As can be seen, the publication of SDEE papers with focus on categorical data is characterized by discontinuity. In fact, no paper was identified in some specific years (1998, 2000, 2003, 2005, 2014, 2017, 2018). Handling categorical data in SDEE has gained research interest in the period 2008-2013 (59% of the selected papers). Outside this period, poor number of studies was identified (not more than one paper per year except for 2001).

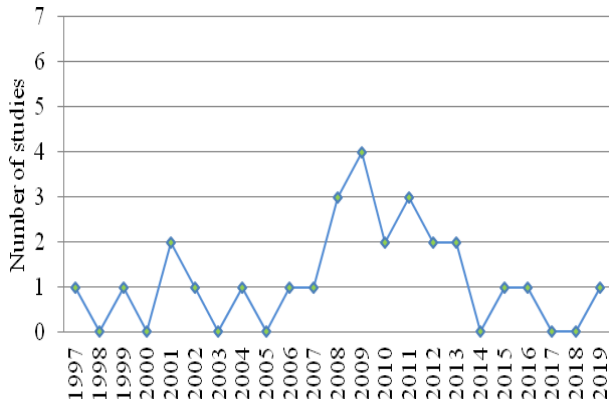


Fig. 2 Publication trends of the selected studies

### D. Research approaches (MQ3) and contribution types (MQ4)

As shown in Fig. 3, two main research approaches were used in the selected papers: solution proposal, and history-based evaluation. The solution proposal approach was adopted by 85% of the selected studies. Among them, 91% (21 out of 23) proposed new techniques, 4% (1 out of 23) proposed a new framework and 4% investigated the use of a new metric. Note that, all selected studies were included in the history-based evaluation approach. Among them, 15% (4 out of 27) performed a comparison of various SDEE techniques using datasets with mixed numerical and categorical data. The remaining papers used historical datasets to assess the performance of their proposed approaches.

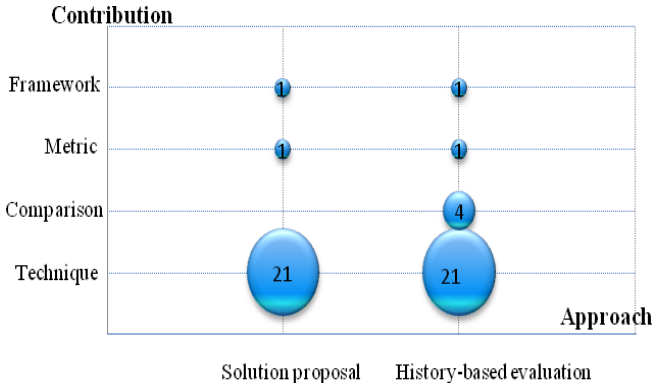


Fig. 3 Research approaches used in the selected studies and their contribution type

### E. SDEE Techniques investigating categorical data (MQ5)

Various approaches were used in the selected papers to estimate software effort using a mixture of numerical and categorical data. Table V shows the techniques used as well as the number of studies in which they were applied. Case based-reasoning (CBR), Regression (SR), Fuzzy Logic (FL), and Classification and Regression Trees (CART) were the techniques that investigate the most the use of categorical data in software effort estimation. Most of these techniques were not used alone. They were combined with each other to improve their prediction accuracy and to get accurate estimates. Specifically, 59% (16 out of 27) of the selected papers used a combination of two or more techniques to predict software effort whereas 41% employed a single technique.

TABLE V. TECHNIQUES USED IN THE SELECTED PAPERS

| Technique used | # of studies | Studies |
|---|---|---|
| Case based-reasoning (CBR) | 15 | S3, S6, S7, S8, S9, S10, S12, S13, S15, S16, S17, S18, S19, S20, S24 |
| Regression (SR) | 9 | S1, S4, S12, S16, S20, S22, S25, S26, S27 |
| Fuzzy Logic (FL) | 7 | S2, S3, S8, S9, S14, S15, S21 |
| Classification and Regression Trees (CART) | 5 | S11, S12, S14, S16, S21 |
| Model Tree (MT) | 2 | S5, S7 |
| Artificial Neural Networks (ANN) | 2 | S19, S26 |
| Grey Relational Analysis (GRA) | 2 | S8, S9 |
| Stepwise ANOVA | 2 | S12, S16 |
| Bees Algorithm (BA) | 1 | S5 |
| Kendall's Row-wise Rank Correlation | 1 | S6 |

| | | |
|---|---|---|
| (CRRC) | | |
| Particle Swarm Optimization (PSO) | 1 | S10 |
| Association Rules (AR) | 1 | S11 |
| Mantel's Correlation (MC) | 1 | S17 |
| Collaborative Filtering (CF) | 1 | S18 |
| Genetic Programming (GP) | 1 | S23 |
| IFPUG | 1 | S25 |

### *F. Handling of categorical data in SDEE (MQ6)*

To deal with categorical data, different techniques were applied depending on their type (nominal or ordinal) as well as the SDEE technique in which they were used. Table VI shows how both nominal and ordinal data were handled in the selected SDEE studies. Note that, some studies used the term 'Categorical' without specifying the exact data type (nominal or ordinal). As shown in table VI, using CBR, Euclidean distance is the most used metric to assess the similarity between two projects that are described by a mixture of numerical and categorical data [9], [27]-[33]. Fuzzy logic, and fuzzy clustering techniques were also used in many CBR/DT works to deal with categorical data [10], [17], [20], [34], [35]. Using regression, most papers employed one-way Analysis of Variance (ANOVA) and recorded categorical variables into new ones with fewer categories [2], [18], [30], [31], [36], [37]. Other studies employed classification and regression trees to handle categorical data [30], [31], [35], [38]-[40].

The above-mentioned techniques were applied to handle both nominal and ordinal data. Other techniques to deal with categorical data were identified depending on whether they are measured on a nominal or ordinal scale. Table VII shows how nominal data were handled in the selected papers. Using regression, four techniques were identified: Transformation to dummy variables, dataset segmentation, interaction, and use of a hierarchical linear model [30], [31], [36], [41], [42]. Using CBR, the equality distance was used to assess the similarity between projects that are described by nominal features [1], [36]. Regarding ordinal data, they were handled as if they were measured using an interval scale or converted to numerical values using regression [30], [43]. Using CBR, they were treated as interval scaled or handled using Grow's formula [1], [36] (see table VIII).

It is worth noting that, when investigating the use of categorical data in the selected papers, we found that some CBR works used categorical data not only to measure the similarity between software projects using Euclidean distance but also: 1) to adjust estimation by analogy; 2) to identify whether a categorical attribute is appropriate to yield predictions or 3) for feature weighting (see table IX).

TABLE VI. CATEGORICAL (NOMINAL AND ORDINAL) DATA HANDLING

| Categorical data handling | SDEE Technique | Studies |
|---|---|---|
| Euclidean distance | CBR | S6, S7, S10, S12, S16, S17, S19, S24 |
| Combining categories / ANOVA | Regression | S4, S12, S16, S20, S22, S26 |
| Classification by DT | Decision trees | S5, S11, S12, S14, S16, S21 |
| Fuzzy logic | CBR / DT | S2, S3, S13, S15, S14 |
| Fuzzy Clustering technique | CBR | S3, S13, S15 |
| Quantification of data | Regression | S4 |
| Grey Relational Coefficient | CBR | S8 |
| Manhattan distance | CBR | S10 |
| Local similarity | CBR | S18 |
| Grammar Guided Genetic Programming | Genetic Programming | S23 [44] |

Table VII. Nominal data handling

| Nominal data handling | SDEE Technique | Studies |
|---|---|---|
| Dummy variables | Regression | S12, S16, S20, S27 |
| Equality distance | CBR | S9, S20 |
| Dataset segmentation | Regression | S25, S27 |
| interaction | Regression | S27 |
| hierarchical linear model | Regression | S27 |

Table VIII. Ordinal data handling

| Ordinal data handling | SDEE Technique | Studies |
|---|---|---|
| Interval scale | Regression / CBR | S12, S20 |
| Grow's formula | CBR | S9 |
| Conversion to numerical values | Regression | S1 |

Table IX. Other uses of categorical data

| Use of categorical data | SDEE Technique | Studies |
|---|---|---|
| Adjustment using MT | CBR | S7 |
| Adjustment using ANN | CBR | S19 |
| Weighting using PSO | CBR | S10 |
| Appropriateness of attributes using CORR | CBR | S6 |
| Dataset appropriateness using Mantel's correlation (dataset partitioning based on nominal data) | CBR | S17 |

### *G. Types of used categorical data (MQ7)*

Fig. 4 shows the types of categorical data used in the selected papers. As can be seen, 59% (16 out of 27) of the selected studies dealt with both nominal and ordinal data, 7% (2 out of 27) dealt with only nominal data and 4% (1 out of 27) were concerned with ordinal data. Among the selected studies, 30% (8 out of 27) did not specify the exact

categorical data type that is handled in the paper. However, based on our knowledge and the datasets used in the experiments, we concluded that most of these papers dealt with both nominal and ordinal data types.

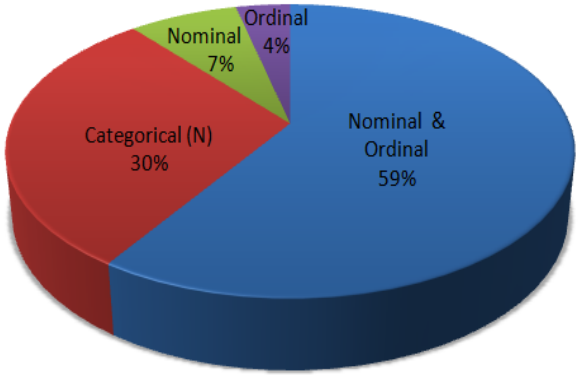


Fig. 4 Types of used categorical data

### *H. Datasets used (MQ8)*

Several datasets were used in the selected papers to investigate the use of categorical data in software effort estimation. Table X shows the datasets used for validation as well as the number of studies in which they were used and the percentage of categorical data. The min, max, and mean columns show the minimum value, the maximum value and the mean value respectively of the percentage of categorical data used in the selected papers to conduct experiments. Note that, different studies may opt for different categorical features to conduct experiments. Therefore, the percentage of categorical data is not the same for all studies. Note also that, there were some studies for which it was not possible to extract the percentage of categorical features used in the experiments. As can be seen from table X, 21 datasets were used in the selected papers. Among them ISBSG, COCOMO, Desharnais, Kemerer, Albrecht and Maxwell are the most used datasets. In terms of categorical data percentage, COCOMO (93.52%) was the dataset with the highest mean percentage followed by Maxwell (88.83%) and Laturi (80.00%).

Even if ISBSG is the most used dataset and contains numerous categorical features, the mean percentage of the categorical data used in the selected papers was 49.13%. This is due to the fact that some studies used few categorical features to conduct experiments. Also, there was 1 study [29] that used only the numerical features of ISBSG. This study was included in our mapping study since the technique described in the paper may be applied on both numerical and categorical data. It is worth noting that, some papers employed datasets with numerical and mixed data to show the efficiency of their techniques to deal with both data types.

Table X. Datasets used in the selected papers

| Dataset | # of studies | Percentage of categorical data | | |
|---|---|---|---|---|
| | | Min | Max | Mean |
| ISBSG | 19 | 00.00 | 81.82 | 49.13 |
| COCOMO | 11 | 88.89 | 94.74 | 93.52 |
| Desharnais | 9 | 10.00 | 25.00 | 13.10 |
| Kemerer | 6 | 33.00 | 40.00 | 36.58 |
| Albrecht | 6 | 00.00 | 16.67 | 8.33 |
| Maxwell | 5 | 80.00 | 95.65 | 88.83 |
| NASA93 | 3 | 60.00 | 94.44 | 77.22 |
| Telecom | 2 | N | N | N |
| USP05-FT | 2 | 52.94 | 63.64 | 58.29 |
| USP05-RQ | 2 | 52.94 | 63.64 | 58.29 |
| China | 1 | 00.00 | 00.00 | 00.00 |
| DPS | 1 | 00.00 | 00.00 | 00.00 |
| CF | 1 | 00.00 | 00.00 | 00.00 |
| STTF | 1 | 15.62 | 15.62 | 15.62 |
| Laturi | 1 | 80.00 | 80.00 | 80.00 |
| Leung02 | 1 | 00.00 | 00.00 | 00.00 |
| Mends03 | 1 | 00.00 | 00.00 | 00.00 |
| Atkinson | 1 | N | N | N |
| Finnish | 1 | N | N | N |
| Mermaid | 1 | N | N | N |
| Real-time 1 | 1 | 100.00 | 100.00 | 100.00 |

N: Not given in the paper

## IV. IMPLICATION FOR RESEARCH AND PRACTICE

This study aims at presenting an overview of how categorical data are handled in SDEE. Based on the finding of our SMS, some recommendations to SDEE researchers and practitioners are provided. Dealing with categorical data is an important issue in SDEE especially at the early stages of the software life cycle where most of the existing attributes are more categorical than numerical. This study found that, the publication of SDEE papers with focus on categorical data is characterized by discontinuity. This implies that the use of categorical data in SDEE needs to be more investigated.

No case study was identified in the selected papers. Therefore, it is suggested to the researchers to cooperate with practitioners in order to explore the use of categorical data in industry to yield estimates. We also recommend for researchers to develop tools that enable software effort estimation using a mixture of numerical and categorical data to encourage the use of categorical data by practitioners and researchers.

This study found that CBR, regression and classification and regression trees are the techniques that investigate the most the use of categorical data in SDEE. It is therefore recommended to conduct further research works using other SDEE techniques. Researchers are also encouraged to develop new techniques to handle categorical data instead of

using traditional ones. Furthermore, previous studies revealed that ensemble techniques yield better results than single techniques [26], [45]-[47]. However, all selected papers used single SDEE techniques. No ensemble SDEE technique dealing with categorical data was identified. This implies that researchers should give more attention to the use of categorical data in ensemble techniques to investigate their impact on improving the estimation accuracy of their techniques.

## V. Conclusion and Future Work

In this paper, a systematic mapping study was carried out in order to identify and summarize the existing works on SDEE dealing with categorical data. A total of 27 relevant studies were identified and classified according to research approach, contribution type, SDEE technique, Technique used to handle categorical data, types of categorical data and datasets used. Research sources and publication trends were also identified and analyzed. Our findings are summarized as follows.

**(MQ1):** Dealing with categorical data has not been sufficiently investigated in SDEE. Besides, Journals were the most targeted publication channels followed by conferences.

**(MQ2):** The publication of SDEE papers with focus on categorical data is characterized by discontinuity. Dealing with categorical data in SDEE has gained research interest in the period 2008-2013.

**(MQ3):** Solution proposal and history-based evaluation were the two main research approaches used in the selected papers.

**(MQ4):** Most of the selected papers focus on developing new techniques especially to improve existing approaches.

**(MQ5):** Case based-reasoning, regression, fuzzy logic, and classification and regression trees were the techniques that investigate the most the use of categorical data in SDEE.

**(MQ6):** Euclidean distance, fuzzy logic, and fuzzy clustering techniques were the most used techniques to handle categorical data using CBR. Using regression, most papers employed ANOVA and combination of categories.

**(MQ7):** Most of the selected studies dealt with both nominal and ordinal data.

**(MQ8):** ISBSG, COCOMO, Desharnais, Kemerer, Albrecht and Maxwell were the most used datasets.

For future work, we will carry out a systematic literature review to analyze the use of categorical data in SDEE by taking into account the finding of this SMS.

## References


[1] M. Azzeh, D. Neagu, and P. Cowling, "Software effort estimation based on weighted fuzzy grey relational analysis", in *Proc. 5th International Workshop on Predictive Models in Software Engineering*, Vancouver, BC, Canada, 2009. https://doi.org/10.1145/1540438.1540450. S9*

[2] I.F. de Barcelos Tronto, J.D. S. da Silva, and N. Sant'Anna, "An investigation of artificial neural networks based prediction systems in software project management", *The Journal of Systems and Software*, vol. 81, pp. 356–367, 2008. https://doi.org/10.1016/j.jss.2007.05.011. S26*

[3] B.W. Boehm, "Software cost estimation with COCOMOII", NJ: Prentice-Hall, 2000.

[4] E. Mendes, "The use of Bayesian networks for Web effort estimation: further investigation", in *Proc. 8th Int Conf on Web Engineering*, New York, 2008, pp. 203–216. https://doi.org/10.1109/ICWE.2008.16.

[5] J. Wen, S. Li, Z. Lin, Y. Huc, and C. Huang, "Systematic literature review of machine learning based software development effort estimation models", *Information and Software Technology*, vol. 54, no. 1, pp. 41–59, 2012. https://doi.org/10.1016/j.infsof.2011.09.002.

[6] S.-J. Huang, N.-H. Chiu, and L.-W. Chen, "Integration of the grey relational analysis with genetic algorithm for software effort estimation", *European Journal of Operational Research*, vol. 188, no. 3, pp. 898–909, 2008. https://doi.org/10.1016/j.ejor.2007.07.002.

[7] K.V. Kumar, V. Ravi, M. Carr, and N.R. Kiran, "Software development cost estimation using wavelet neural networks", *Journal of Systems and Software*, vol. 81, pp.1853–1867, 2008. https://doi.org/10.1016/j.jss.2007.12.793.

[8] M.O. Elish, "Improved estimation of software project effort using multiple additive regression trees", *Expert Systems with Applications*, vol. 36, no. 7, pp. 10774–10778, 2009. https://doi.org/10.1016/j.eswa.2009.02.013.

[9] M. Shepperd and C. Schofield, "Estimating software project effort using analogies", *IEEE Transactions on Software Engineering*,vol. 23, no. 12, pp. 736–743, 1997. https://doi.org/10.1109/32.637387. S24*

[10] M.A. Ahmed and Z. Muzaffar, "Handling imprecision and uncertainty in software development effort prediction: a type-2 fuzzy logic based framework", *Information and Software Technology*, vol. 51, no. 3, pp. 640–654, 2009. https://doi.org/10.1016/j.infsof.2008.09.004. S2*

[11] R.T. Hughes, "Expert judgment as an estimating method", *Information and Software Technology*, vol. 38, pp. 67–75, 1996. https://doi.org/10.1016/0950-5849(95)01045-9.

[12] A. Idri, T. Khoshgoftaar, and A. Abran, "Investigating soft computing in case-based reasoning for software cost estimation", *Inter. Jour. of Eng. Int. Sys. for Ele. Eng. and Com*., vol 10, no. 3, pp. 147-157, 2002.

[13] A. Idri, A. Abran, and L. Kjiri, "COCOMO Cost Model Using Fuzzy Logic", in *Proc. 7th International conference on Fuzzy Theory and technology*, Atlantic, New Jersy, 2000, pp. 1–4.

[14] B. Boehm, "Software engineering economics", *IEEE Transactions on Software Engineering*, vol. 10 pp. 4–21, 1984. https://doi.org/10.1109/TSE.1984.5010193.

[15] ISBSG, International Software Benchmark and Standard Group, www.isbsg.org.

[16] A. Idri, A. Abran, and T. Khoshgoftaar, "Fuzzy Analogy: a New Approach for Software Effort Estimation", in Proc. *11th International Workshop in Software Measurements*, Canada, 2001, pp. 93-101.

[17] F.A. Amazal, A. Idri, and A. Abran, "Improving Fuzzy Analogy Based Software Development Effort Estimation", in *Proc. 21st Asia-Pacific Software Engineering Conference*, Jeju, South Korea, 1-4 Dec, 2014. https://doi.org/10.1109/APSEC.2014.46. S3*

[18] L. Angelis, I. Stamelos, and M. Morisio, "Building a Software Cost Estimation Model Based on Categorical Data", in *Proc. 7 th International Software Metrics Symposium*, London, UK, 2001, pp. 4–15. https://doi.org/10.1109/METRIC.2001.915511. S4*

[19] M. Azzeh, D. Neagu, P. Cowling, "Fuzzy grey relational analysis for software effort estimation", *Empirical Software Engineering*, vol. 15, no. 1, pp 60–90, 2010. https://doi.org/10.1007/s10664-009-9113-0. S8*

[20] A. Idri, F.A. Amazal, and A. Abran, "Accuracy Comparison of Analogy-Based Software Development Effort Estimation Techniques", *International Journal of Intelligent Systems*, vol. 31, no. 2, pp. 128-152, February 2016. https://doi.org/10.1002/int.21748. S15*

[21] J. Li, G. Ruhe, A. Al-Emran, and M. Richter, "A flexible method for software effort estimation by analogy", *Empirical Software Engineering*, vol. 12, pp. 65–106, 2007. https://doi.org/10.1007/s10664-006-7552-4. S18*

[22] B. Kitchenham, D. Budgen, and O.P. Brereton, "The value of mapping studies – A participant-observer case study", in Proc. 14th International Conference on Evaluation and Assessment in Software Engineering, Keele University, UK, 2010, pp. 1–9.

[23] B. Kitchenham, O.P. Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering – A systematic literature review", Information and Software Technology, vol.51, pp. 7–15, 2009. https://doi.org/10.1016/j.infsof.2008.09.009.

[24] B. Kitchenham, S. Charters, "Guidelines for Performing Systematic Literature Reviews in Software Engineering", Tech. Rep. EBSE-2007-01, Keele University and University of Durham, 2007.

[25] A. Idri, F.A. Amazal, and A. Abran, "Analogy-based software development effort estimation: A systematic mapping and review", Information and Software Technology, vol. 58, pp.206–230, 2015. https://doi.org/10.1016/j.infsof.2014.07.013.

[26] A. Idri, M. Hosni, and A. Abran, "Systematic Mapping Study of Ensemble Effort Estimation", in Proc. 11th International Conference on Evaluation of Novel Software Approaches to Software Engineering, 2016, pp. 132–139. https://doi.org/10.5220/0005822701320139.

[27] M. Azzeh, "Dataset Quality Assessment: An extension for analogy based effort estimation". International Journal of Computer Science & Engineering Survey (IJCSES), vol.4, no.1, 2013. S6*

[28] M. Azzeh, "Model tree based adaption strategy for software effort estimation by analogy", in Proc. of the 11th International Conference on Computer and Information Technology, Pafos, Cyprus, 2011. https://doi.org/10.1109/CIT.2011.48. S7*

[29] V. K. Bardsiri, D. N. A. Jawawi, S. Z. M. Hashim, and E. Khatibi, "A PSO-based model to increase the accuracy of software development effort estimation", Software Quality Journal, vol. 21, no. 3, pp. 501-526, 2013. https://doi.org/10.1007/s11219-012-9183-x. S10*

[30] L. C. Briand, K. El Emam, D. Surmann, I. Wieczorek, and K. D. Maxwell, "An Assessment and Comparison of Common Software Cost Estimation Modeling Techniques", in Proc. of the International Conference on Software Engineering (ICSE), Los Angeles, CA, USA , May 1999. https://doi.org/10.1145/302405.302647. S12*

[31] R. Jeffery, M. Ruhe, and I. Wieczorek, "Using Public Domain Metrics to Estimate Software Development Effort", in Proc. 7th International Symposium on Software Metrics, April 04 - 06, 2001. https://doi.org/10.1109/METRIC.2001.915512. S16*

[32] J. W. Keung, B. Kitchenham, and D. R. Jeffery, "Analogy-X: Providing Statistical Inference to Analogy-Based Software Cost Estimation", IEEE Transactions on Software Engineering, vol. 34, no. 4, July/August 2008. https://doi.org/10.1109/TSE.2008.34. S17*

[33] Y. F. Li, M. Xie, and T. N. Goh, "A study of the non-linear adjustment for analogy based software cost estimation", Empirical Software Engineering, vol. 14, no. 6, pp. 603–643, December 2009. https://doi.org/10.1007/s10664-008-9104-6. S19*

[34] L. Haitao, W. Ru-xiang, and J. Guo-ping, "Similarity measurement for data with high-dimensional and mixed feature values through fuzzy clustering", in Proc. International Conference on Computer Science and Automation Engineering (CSAE), 2012. https://doi.org/ 10.1109/CSAE.2012.6273028. S13*

[35] S.-J Huang, C.-Y Lin, and N.-H Chiu, "Fuzzy Decision Tree Approach for Embedding Risk Assessment Information into Software Cost Estimation Model", Journal of Information Science and Engineering, vol. 22, pp. 297-313, 2006. S14*

[36] N. Mittas, and L. Angelis, "LSEbA: least squares regression and estimation by analogy in a semi-parametric model for software cost estimation", Empirical Software Engineering, vol.15, pp. 523–555, 2010. https://doi.org/10.1007/s10664-010-9128-6. S20*

[37] P. Sentas, L. Angelis, I. Stamelos, and G. Bleris, "Software productivity and effort prediction with ordinal regression", Information and Software Technology, vol. 47, pp. 17–29, 2005. https://doi.org/10.1016/j.infsof.2004.05.001. S22*

[38] M. Azzeh, "Software Effort Estimation Based on Optimized Model Tree", in Proc. 7th International Conference on Predictive Models in Software Engineering, Banff, Alberta, Canada, September 20-21, 2011. S5*

[39] S. Bibi, I. Stamelos, and L. Angelis, "Combining probabilistic models for explanatory productivity estimation", Information and Software Technology, vol. 50, pp. 656–669, 2008. https://doi.org/10.1016/j.infsof.2007.06.004. S11*

[40] E. Papatheocharous, and A. S. Andreou, "Classification and Prediction of Software Cost through Fuzzy Decision Trees". in Proc. International Conference on Enterprise Information Systems, 2009, pp. 234-247. https://doi.org/10.1007/978-3-642-01347-8_20. S21*

[41] M. Tsunoda, S. Amasaki, and A. Monden, "Handling categorical variables in effort estimation", in Proc. 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, 20-21 Sept. 2012. https://doi.org/10.1145/2372251.2372267. S27*

[42] P. Silhavy, R. Silhavy, and Z. Prokopova, "Categorical Variable Segmentation Model for Software Development Effort Estimation", IEEE Access, vol. 7, pp. 9618 - 9626, 11 January 2019. https://doi.org/10.1109/ACCESS.2019.2891878.S25*

[43] R. Abdukalykov, I. Hussain, M. Kassab, and O. Ormandjieva, "Quantifying the Impact of Different Non-functional Requirements and Problem Domains on Software Effort Estimation", in Proc. Ninth International Conference on Software Engineering Research, Management and Applications, Baltimore, MD, USA, 2011. https://doi.org/10.1109/SERA.2011.45. S1*

[44] Y. Shan, R. I. McKay, C.J. Lokan, and D.L. Essam, "Software Project Effort Estimation Using Genetic Programming", in Proc. International Conference on Communications, Circuits and Systems and West Sino Expositions (ICCCAS), Chengdu, China, 29 June-1 July 2002. https://doi.org/10.1109/ICCCAS.2002.1178979. S23*

[45] A. Idri, M. Hosni, and A. Abran, "Systematic Literature Review of Ensemble Effort Estimation", Journal of Systems and Software, vol. 118, pp. 151–175, 2016. https://doi.org/10.1016/j.jss.2016.05.016.

[46] M. Hosni, A. Idri, and A. Abran, "Investigating Heterogeneous Ensembles with Filter Feature Selection for Software Effort Estimation", in Proc. 27th International Workshop on Software Measurement and 12th International Conference on Software Process and Product Measurement, ACM, New York, NY, USA, 2017: pp. 207–220. https://doi.org/10.1145/3143434.3143456.

[47] M. Azzeh, A.B. Nassif, and L.L. Minku, "An empirical evaluation of ensemble adjustment methods for analogy-based effort estimation", Journal of Systems and Software, vol. 103, pp. 36–52, 2015. https://doi.org/10.1016/j.jss.2015.01.028.