International Journal of Computers and Informatics (IJCI)
Vol. (2), No. (1)

IJCI

المجلة الدولية للحاسبات والمعلوماتية
الإصدار (2)، العدد (1)

March 2023

A Proposed Approach for Effort Estimation of Developing Mobile Applications

Karim Sayed Roshdi
M.Sc. of Information Systems, Ain Shams University, Egypt

Ezzat Samy Elbagoury
M.Sc. of Computer Science, Ain Shams University, Egypt

Abstract

Effort estimation techniques play a key role in the planning process for the development of mobile phone applications. The development of mobile applications is different from the traditional applications of information systems because of their dissimilar characteristics and the rapid advancement of technology used in the development of the former. For this, existing traditional effort estimation techniques may not be suitable for use in predicting the development effort of mobile applications. The process of estimating and predicting the effort depends mainly on the characteristics of the applications. The aim of this study is to propose a methodology for the use of intelligent techniques to predict the effort to develop mobile applications, which are considered unconventional, and to cope with the rapid development of the mobile application development environment.

Keywords: Effort Estimation, Mobile Application, Mobile Computing, Systematic Review.

1- Introduction

Nowadays, mobile devices are the fastest-growing computing platform. This rapid explosion of mobile devices over the last five years has dramatically altered the platform that is utilized for social, business, entertainment, gaming, and marketing

using software applications. The development of mobile applications that provide rich content to users by using global positioning sensors (GPS), wireless connectivity, photo/video capabilities, built-in web browsers, and voice recognition, among other sensors became more complex and challenging. New features in mobile devices that did not exist previously in traditional software applications represent a challenge and novel requests for software engineers.

Another challenge added to the estimation of a mobile application is the different kinds of Mobile applications. It falls broadly into three categories: native, web-based, and hybrid Native applications run on a device's operating system and are required to be adapted for different devices. Web-based applications require a web browser on a mobile device. Hybrid applications are 'native-wrapped' web applications. Effort estimation is a vital project management activity needed for project planning[1]. The estimation process is expecting how much effort is needed to develop a software project and maintain based on vague customer requests[2].

For traditional software, several approaches have been defined to support the task that can be divided into two main categories, namely the non-model-based and model-based methods [2]. Broadly speaking, non-model-based methods involve the judgment of human experts, who provide an estimate based on their previous knowledge and practices [3]. Alternatively, model-based approaches rely on the definition of a set of cost drivers used as independent variables in prediction models aimed at estimating a numerical variable. One of the advantages of model-based approaches is that they are more applicable [4]. However, they critically depend on the identification and evaluation of cost drivers. Different approaches devised based on the employed cost drivers and each one can be applied in a different phase of the development process, once the information to evaluate the required cost drivers is available. New approaches specially design models to fit for mobile application effort estimation [5] or adapt existing one [6].

The main objective of this paper is to fetch different and critical factors that affect mobile application development using a systematic literature review (SLR), review previous work of other researchers, compare their work to point out the gaps, and propose an approach model to estimate the effort for mobile application development. to solve those problems.

The remaining part of this paper is organized as follows: Section 2 describes the review process. Section 4 reports the review results. Section 5 summarizes the main recommendations for future research on software estimation for mobile application.

2- Systematic Literature Review

A systematic literature review is conducted methodically by following a set of guidelines to collect and analyze all available evidence about a specific question in an unbiased and repeatable manner [7]. Firstly, following research questions were framed to guide the SLR:

RQ1: What are the characteristics of Mobile Applications?

RQ2: What are the techniques that have been used for effort or size estimation for mobile software development?

To find relevant studies to answer our research questions, the researchers conducted a search composed of two steps. The first step was to define a search string. The second step was to apply this search string to a set of selected digital libraries to extract all the relevant papers. These two steps are described in detail below.

2-1 Search terms

The complete set of search terms was formulated as follows: (mobile OR mob* OR “cell phone” OR telephone) AND (software OR system OR application OR app* OR development) AND (effort OR cost OR resource) AND (estimat* OR predict* OR assess*).

2-2 Literature resource

To answer the research questions, the researchers performed an automated search based on the pre-constructed search terms using the following electronic databases:

– IEEE Digital Library.

– ACM Digital library.

2-3 Search process

As part of the search strategy, search strings were applied on different databases to fetch the relevant studies since 2010 and then remove duplicate articles.

2-4 Study Selection Procedure

The aim of this step was to identify the relevant studies that addressed the research questions based on their title, abstract, and keywords.

To achieve this, the researchers evaluated each of the candidate papers identified in the initial search stage using the inclusion and exclusion criteria, to determine whether it should be retained or rejected. If this decision could not be made using its title and/or its' abstract alone, the full paper was reviewed.

Inclusion criteria:

- Relevant to the topic of our review
- Papers that have been published in the last eight years (2010 - 2017)
- Studies reported in a conference, journal, and thesis OR are reported in a technical report
- Studies wrote in English.

Exclusion criteria:

- Studies that do not present effort estimation models/methods/metrics for mobile application.

- Studies not written in English.

3- ASSESSMENT of Papers

The study quality assessment can be used to guide the interpretation of the synthesis findings and to determine the strength of the elaborated inferences. Nominated paper or articles quality assessed according to score given accord the answer three questions as shown in tabel1. Quesstion1 (Q1) assess if the authors of the study clearly state the aims and objectives of the carried-out research. This question could be answered positively for all the reviewed publications. Question two (Q2) ask if the study provides enough information (either directly or by reference to the relevant literature) to give the presented research the appropriate context and background. Question three (Q3) check if the validity of the model or metrics clearly discussed.

4- Related Work

Jošt et al [8] examine whether the traditional software metrics are appropriate for measuring the mobile applications' source code. the authors introduced eight metrics to be applied in the analysis of the mobile application. However، this method cannot be used in the early stage of development since it relies on the complete source code to get a proper measure. The authors also do not provide additional guidelines on how the proposed metric is applied to the mobile prototype and how they valued (high or low) the source code.

In order to find out which software effort estimation model is more accurate Arnuphaptrairong et al [9] empirically validate and compare the accuracy between Function Points Analysis method and a proposed technique especially design for mobile application effort estimation. The results show disappointed accuracy for both estimation models. The author explains the reason for this poor result to the productivity rates used in both models was very rigid.

Nitze [10] uses the COSMIC Function Points approach to measure the size of mobile application based on its functional user requirements and adapt it by introducing 19 elements for the mobile development. Although, the author concludes that COSMIC is an appropriate approach for mobile application size measurement.it still did not consider non-functional requirements.

Tunal [11] uses Functional Point Analysis (FPA) measurement to estimate the size of mobile tablet application developed by him. The estimate was compared to the actual size of the project after development. Results show high accuracy and low deviation between actual and estimated size. However, there are a variety of mobile applications such as mobile games, which has more complex features and characteristics. The FPA measurement might be not applicable for estimate it.

Shahwaiz et al [12] propose a parametric model to estimate effort needed to develop mobile application cations to help project managers. The authors identified 20 cost drivers initially، choose only 7 of them who has the most impact on the effort of development of mobile applications and then use step forward regression to derive the effort estimation model. The model tested and validated using data of more than 160 mobile projects developed by either freelancer or software houses. The authors claimed that their model outperformed the general-purpose COCOMO Ⅱ. However, they did not test the model for different mobile application types or sizes.

De Souza et al [13] present a proposal for an effort estimation model for mobile applications. They at first identifying specific characteristics of mobile systems and then adapt of an estimation method that exists Finnish Software Measurement Association (FiSMA). However, the authors did not discuss how the measurement is validated.

Abdullah et al [14] proposed UML-based functional measurement through COSMIC function point to be used to estimate the mobile game application. The authors used

the angry birds game as a case study. The key idea in this study is to use UML representations for capturing the information needed for the measurement and estimation.

Ferrucci et al [15] analyze the use of COSMIC Function Points to measure the size of mobile applications and investigate whether the results correlate with code-based measurements. The authors claimed that the COSMIC functional size evaluated was well correlated to all the size measures considered. However, the prediction accuracy was not good compared to other methods.

5- Mobile Application Factors

Some aspects are differentiating mobile applications from the traditional application. These factors divided into three parts, hardware, software, and the communication.

5-1 Hardware Factors

- Limited power: Mobile applications have less processing power and relativity small by the LOC. In addition, mobile applications have less memory space than the traditional application [16].
- Screen size: Mobile phone screen size is small and varies from one device to others.

Table 1: Papers Quality Assessment

Paper ID	Title	References	Q1	Q2	Q3	Score
1	A parametric effort estimation model for mobile apps	(Shahwaiz, Malik, & Sabahat, 2017)	1	0.5	1	2.5
2	A Set of Metrics for the Effort Estimation of Mobile Apps	(Catolino, Salza, Gravino, & Ferrucci, 2017)	1	1	0.5	2.5
3	An analogy-based effort estimation approach for mobile application development projects	(Nitze et al., 2014)	1	1	0.5	2.5
4	An Empirical Validation of Mobile Application Effort Estimation Models	(Arnuphaptrairong & Suksawasd, 2017)	1	1	1	3

5	An Estimation Model for Test Execution Effort Motorola Industrial Ltda	(Aranha, Km, & Brazil, n.d.)	0.5	0.5	0	1
6	An investigation of the accuracy of code and process metrics for defect prediction of mobile applications	(Kaur, Kaur, & Kaur, 2015)	0.5	0.5	0.5	1.5
7	COSMIC functional measurement of mobile applications and code size estimation	(D'Avanzo, Ferrucci, Gravino, & Salza, 2015)	1	1	1	3
8	Estimating the Effort of Mobile Application Development	(De Souza & De Aquino Jr, 2014)	1	0.5	0	1.5
9	Investigating functional and code size measures for mobile applications	(Ferrucci, Gravino, Salza, & Sarro, 2015)	1	0.5	1	2.5
10	Mobile applications, function points and cost estimating	(Preuss, 2013)	0.5	0.5	0.5	1.5
11	Mobile Game Size Estimation	(Abdullah, Rusli, & Ibrahim, 2014)	1	1	1	3
12	On the Use of Requirements Measures to Predict Software Project and Product Measures in the Context of Android Mobile Apps: A Preliminary Study	(Francese, Gravino, Risi, Scanniello, & Tortora, 2015)	0.5	0.5	0.5	1.5

Table 1: Papers Quality Assessment (continue)

Paper ID	Title	References	Q1	Q2	Q3	Score
13	Reviews on functional size measurement in mobile application and UML model	(Atiqah, Abdullah, Ida, & Rusli, 2015)	1	1	0.5	2.5
14	Sizing android mobile applications	(Guruprasath, 2011)				
15	Measure the functional size of a mobile app: Using the cosmic functional size measurement method	(Heeringen & Gorp, 2014)	1	1	1	3
16	Software Size Estimation Using Function Point Analysis – A Case Study for a Mobile Application	(Tunalı, 2014)	1	1	0.5	2.5
17	A case study in COSMIC functional size measurement: Angry Bird Mobile Application	(Abdullah, Rusli, & Ibrahim, 2013)	1	1	1	3
18	Mobile applications, function points and cost estimating	(Preuss, 2013)	1	1	0	2

- Start-up time: Mobile device's user uses the mobile phone in short duration and mobile phone should have an ability to quickly start a mobile application.

5-2 Software Factors

- User-Interface: Mobile application designed to match the target mobile environment. The target environment standards are important to the user with the pleasant application.
- Interaction with the information sources: In the mobile application, data is transferred from one application to other application [17].

5-3 Communication Factors

- Network communication: Networks are a very important factor for communication from one device to another device. Mobile devices are connected to the internet, GPS system by the network communication [18].

6- Proposed Model

In this section, a proposed model steps based on best practices and up to date practical research findings for building effort models by use of machine learning algorithms presented. Figuer1 demonstrate those steps as follow:

Step 1: Identification of potential effort predictors was the very first step. A set of predictors can be considered from literature review, experts, or other general-purpose models [19].

Step 2: collect real-life mobile applications from trusted sites for open source application such as GitHub [20] or from software houses.

Step 3: data contain irrelevant and redundant features that might decrease the model performance that is a critical issue especially in software engineering data [21]. The

third step is to select a subset of features that have significant or similar impacts on the evaluation target as using all features.

Step 4: Noisy and unreliable data may severely influence the predictive accuracy of machine learning models. Data processing is a critical task in the process of building ML models. Data processed through removing outliers cleaning, re- duction, transformation, and scaling features before building estimation model. Scaling transforms feature values according to a defined rule so that all scaled features have the same degree of influence.

Step 5: Partition of the dataset could perform using cross-validation schema. Cross-validation is a simple but effective way for the parameter tuning scheme [22]. The dataset is randomly split into three equally sized and mutually exclusive subsets: training subset, validating subset, and testing subset. The training subset is used to construct models with specified parameter settings. The validating subset is used for parameter tuning and to prevent the over-fitting problem of ML methods. The testing subset is used to evaluate the predictive abilities of training methods with optimal parameters.

Step 6: Machine Learning (ML) algorithms, the main advantage is adjusting to the changing environment. This is an important factor for software estimation because there are frequent technology advances, new tools, and programming languages available, and methodology improvements and changing skill sets of project teams that affect it [23]. Numerous ML algorithms available applied depending on the desired outcome. In this study, depending on labeled data so the supervised ML algorithm will be implemented. This step repeated and the next step to select the optimal values of tuning parameters that give the best accuracy or at least accepted one.

Step7: Evaluation of the effort estimation model inspected by using the following method.

a) Magnitude of relative error (MRE): MRE is a measure, which calculates the difference between values estimated by a suggesting model and the values estimated.
b) Mean magnitude of relative error (MMRE): MMRE is the mean measurement of the absolute values of the relative errors from the complete data set [24].
c) Root Mean Squared Error (RMSE): RMSE is the average variation between actual and predicted values predicted by the model [4]. RMSE can only be compared between models whose errors are measured in the same units.

The model that has the lower MRE and MMER considered as an acceptable model.

Figure 2 shows the basic black-box structure of our model. As shown in this figure, our model takes estimation predictors as inputs supplied by the user and generates an estimate of the source lines of code (SLOC) as output. The SLOC converted to the functional size by using a conversion function.

Conclusion and Future Work

The effort estimation techniques have a high strength in the planning of mobile application development. In this paper, the researchers used the steps of SLR to present and point out the gaps and future directions for research in effort estimation for mobile application by answering the questions that were given in this field by following the steps of SLR.

SLR steps such as search strategy, search terms, literature resource, search process, study selection, procedures, inclusion criteria, exclusion criteria, quality assessment, data extraction and synthesis finally presented the result of this work.

In this paper the researchers suggest an approach to predict the effort to develop mobile applications using machine-learning technique. The proposed approach consists of seven steps that start with data collection and ends with evaluation of models to select the best one to use.

The future work: the researchers will apply the proposed model in a case study.

References

[1] M. Łab\kedzki, P. Promiński, A. Rybicki, and M. Wolski, "Agile effort estimation in software development projects--case study," Cent. Eur. Rev. Econ. Manag., vol. 1, no. 3, pp. 135–152, 2017.

[2] A. T. ; Raslan, R. D. ; Nagy, and H. A. Hefny, "Towards a Fuzzy based Framework for Effort Estimation in Agile Software Development," Int. J. Comput. Sci. Inf. Secur., vol. 13, no. 1, pp. 37–45, 2015.

[3] P. Pospieszny, B. Czarnacka-Chrobot, and A. Kobylinski, "An effective approach for software project effort and duration estimation with machine learning algorithms," J. Syst. Softw., vol. 137, no. November, pp. 184–196, 2018.

[4] R. Tripathi and P. K. Rai, "Machine Learning Methods of Effort Estimation and I t ' s Performance Evaluation Criteria," vol. 6, no. 1, pp. 61–67, 2017.

[5] A. Nitze, A. Schmietendorf, and R. Dumke, "An analogy-based effort estimation approach for mobile application development projects," Proc. - 2014 Jt. Conf. Int. Work. Softw. Meas. IWSM 2014 Int. Conf. Softw. Process Prod. Meas. Mensura 2014, pp. 99–103, 2014.

[6] L. S. De Souza and G. S. De Aquino, "Mobile application development: How to estimate the effort?," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8583 LNCS, no. PART 5, pp. 63–72, 2014.

[7] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," Engineering, vol. 2, p. 1051, 2007.

[8] G. Jošt, J. Huber, and M. H. E. R. Ičko, "Using Object Oriented Software Metrics for Mobile Application Development," Second Work. Softw. Qual. Anal. Monit. Improv. Appl., pp. 17–27, 2013.

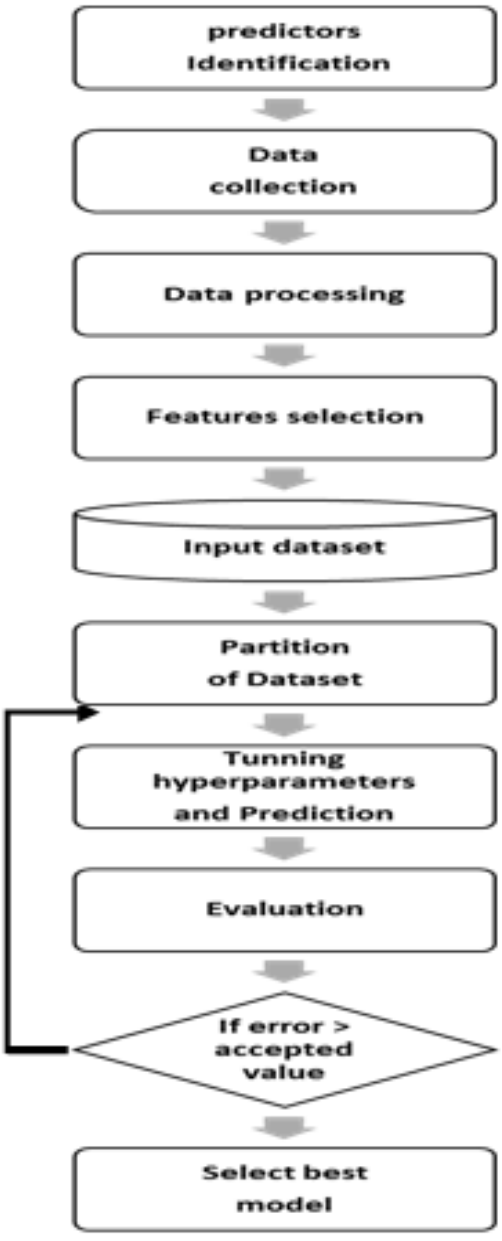

Fig. 1. Proposed Model Steps.

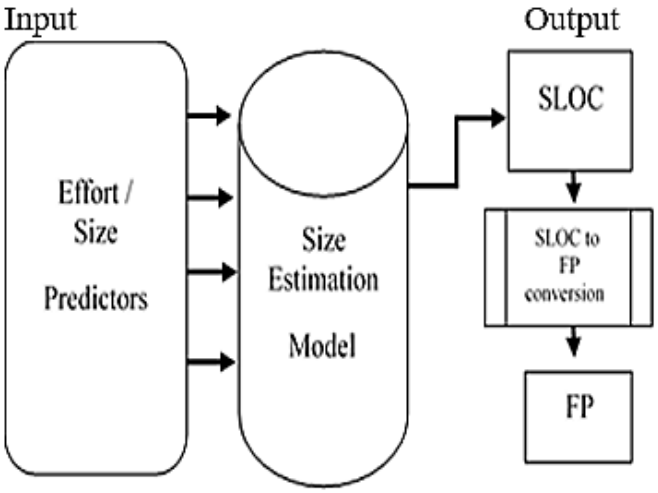

Fig. 2. Size Estimation Model (Black-Box)

[9] T. Arnuphaptrairong and W. Suksawasd, "An Empirical Validation of Mobile Application Effort Estimation Models," vol. II, 2017.

[10] A. Nitze, "Measuring Mobile Application Size Using COSMIC FP," MetriKon, no. March, pp. 101–114, 2013.

[11] V. Tunalı, "Software Size Estimation Using Function Point Analysis – A Case Study for a Mobile Application," 7. Mühendislik ve Teknol. Sempozyumu, no. May, pp. 73–76, 2014.

[12] S. A. Shahwaiz, A. A. Malik, and N. Sabahat, "A parametric effort estimation model for mobile apps," 19th Int. Multi-Topic Conf. INMIC 2016, 2017.

[13] L. S. De Souza and G. S. De Aquino Jr, "Estimating the Effort of Mobile Application Development," Comput. Sci. Inf. Technol. (CS IT), pp. 45–63, 2014.

[14] N. A. S. Abdullah, N. I. A. Rusli, and M. F. Ibrahim, "A case study in COSMIC functional size measurement: Angry bird mobile application," in 2013 IEEE Conference on Open Systems, ICOS 2013, 2013, pp. 139–144.

[15] F. Ferrucci, C. Gravino, P. Salza, and F. Sarro, "Investigating functional and code size measures for mobile applications: A replicated study," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9459, 2015.

[16] H. K. Flora, X. Wang, and S. V.Chande, "An Investigation into Mobile Application Development Processes: Challenges and Best Practices," Int. J. Mod. Educ. Comput. Sci., vol. 6, no. 6, pp. 1–9, 2014.

[17] N. Singh and D. Soni, "Proposing New Model for Effort Estimation of Mobile Application Development," Int. J. Comput. Appl., vol. 170, no. 3, pp. 975–8887, 2017.

[18] R. Francese, C. Gravino, M. Risi, G. Scanniello, and G. Tortora, "On the Use of Requirements Measures to Predict Software Project and Product Measures in the Context of Android Mobile Apps: A Preliminary Study," Proc. - 41st Euromicro Conf. Softw. Eng. Adv. Appl. SEAA 2015, no. ii, pp. 357–364, 2015.

[19] N. Sabahat, A. L. I. A. Malik, and F. Azam, "A Size Estimation Model for Board-Based Desktop Games," vol. 5, 2017.

[20] F. Qi, X. Jing, X. Zhu, X. Xie, B. Xu, and S. Ying, "Software effort estimation based on open source projects : Case study of Github," vol. 92, pp. 145–157, 2017.

[21] S. K. Sehra, Y. S. Brar, N. Kaur, and S. S. Sehra, "Research patterns and trends in software effort estimation," Inf. Softw. Technol., vol. 91, pp. 1–21, 2017.

[22] J. Huang, Y. Li, and M. Xie, "An empirical analysis of data preprocessing for machine learning-based software cost estimation," Inf. Softw. Technol., vol. 67, pp. 108–127, 2015.

[23] P. Pospieszny, B. Czarnacka-chrobot, and A. Kobylinski, "The Journal of Systems and Software An effective approach for software project effort and duration estimation with machine learning algorithms," J. Syst. Softw., vol. 137, pp. 184–196, 2018.

[24] P. Rijwani and S. Jain, "Enhanced Software Effort Estimation Using Multi Layered Feed Forward Artificial Neural Network Technique," Procedia Comput. Sci., vol. 89, pp. 307–312, 2016.