TELKOMNIKA, Vol.16, No.5, October 2018, pp.2082~2090
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v16i5.6561

Effort Estimation Development Model for Web-based Mobile Application Using Fuzzy Logic

Stefani Agusta, Suharjito, Abba Suganda Girsang
Computer Science Department, BINUS Graduate Program-Master of Computer Science,
Bina Nusantara University,Jakarta, 11480, Indonesia
*Corresponding author, e-mail: suharjito@binus.edu[2], agirsang@binus.edu[3]

Abstract

Effort estimation becomes a crucial part in software development process because false effort estimation result can lead to delayed project and affect the successful of a project. This research proposes a model of effort estimation for web-based mobile application developed using object oriented approach. In the proposed model, functional size measurement of object oriented based web application named OOmFPWeb, web metric and mobile characteristic for web-based mobile application size measurement are combnined. The estimation process is done by using mamdani fuzzy logic method. To evaluate the proposed model, the comparison between OOmFPWeb as the variable that affect effort estimation for web-based mobile application and the proposed model are performed. The evaluation result shows that effort estimation for web-based mobile application with the proposed model is better than just using OOmFPWeb.

Keywords: *fuzzy logic, mobile application effort estimation, functional, hypermedia*

1. Introduction

The longer user of smart phones in the world is increasing. Mobile-based application developer is vying to market their creativity to create applications that can support a person needs in a mobile device [1]. Developing a mobile application project need an estimation of successful project. Some criteria of a successful project are completed on time, within budget, and in accordance with the desired quality [2]. It leads to estimate these three factors become very critical. Until now, there are many projects fail due to errors in estimating thes three factors. Effort estimation of software project is difficult because every software has different complexity, high degree of flexibility, and the different abstarction of software. The mobile application is one type of software which has the characteristics and a higher level of difficulty than software effort estimation.

Before estimating the effort needed for project, the project size becomes the important issue. Regarding this issue, the number research of web application is more than web based mobile application. Mendes uses web metric to know the factors of web-based application. This method has limitation which is only for static data web and can’t be used for website which requires dynamic and complex data, like e-commerce, e-learning, and so forth. Mendes then proposed web metrics which can be used for both hypermedia and software web applications size measurement [3]. Web metric is not specific for object oriented based web application. OO-method Function Point for Web (OOmFPWeb) could be used to estimate functional size for web application using an evaluation framework for functional size measurement (FSM) method or procedure by Abrahao. The result of method evaluation has been proved to be efficient for user [4]. But this result of method does not consider multimedia components (audio, animation, image, etc). So, website with same functionality but different interface and different total multimedia elements (audio, animation, image, etc) will be considered to have same effort needed.

Characteristics of mobile application by Souza could be used to know the factor of mobile characteristic. It is concluded that the research presented is entirely appropriate and viable and that this proposal should take into account all the peculiarities of such applications, finally creating a belief that there are considerable differences in the development project for mobile applications [5]. There are efforts to estimate such a method based on expert opinion,

Received May 16, 2018; Revised July 28, 2018; Accepted September 7, 2018

the model algorithm, and based on artificial intelligence. One of which is based on artificial intelligence is fuzzy logic. Fuzzy logic is one of the most commonly method used today. The purpose of this paper is to present the design and evaluation of proposed method for an estimation model for mobile applications. The evaluation of proposed method accuracy is using MMRE [6], MdMRE [7] and Pred(n) [8].

2. Literature Review

Fuzzy logic methods are implemented into many cases to solve the problem [9,10]. Many research about fuzzy logic to effort estimation software development had been done by [11-16]. That research proved that fuzzy logic can be used to estimate the effort in developing software and combining other several methods. Sharma mentioned that the fuzzy logic can help to deal with uncertainty and imprecision in comparison with other popular estimation model [10]. Prasad concluded that using triangular membership function (TMF) is better than fuzzy logic method used GBeIIMF [13]. Gracia ever compared two models of fuzzy logic to estimate software development effort, these two models are models of Takagi-Sugeno fuzzy and Mamdani fuzzy.

The results of the research explained that the Mamdani fuzzy is more accurate for estimating software development effort that project ≤ 100 than Takagi-Sugeno fuzzy [17]. Based on this result, we use Mamdani fuzzy to estimation web-based mobile application development effort. But researches had not yet been made the effort estimation in mobile application, the presence of this literature study, the authors wish to back for the research of fuzzy logic for effort estimation for mobile application development. Research about web metric to determine web size had been done by Cowderoy [18], Mendes [19], Reifer [20], Cleary [21] only used data from one web company, which might effect external validation from their result. Mendes suggested web metrics for hypermedia based web application and Mendes suggested web metric for static and dynamic web application.

3. Research Method

Figure 1 shows the steps of the proposed effort estimation methods. First, data is taken from the web-based mobile application project requirement specification. It will generate the projects size. Then the the estimation model with fuzzy logic is developed. It starts from fuzzification, then uses interface engine which records data estimate project in database and also generates the rule base. The matlab is used as to develop interface engine. Once completed, the stage defuzzification done to get effort results sought.

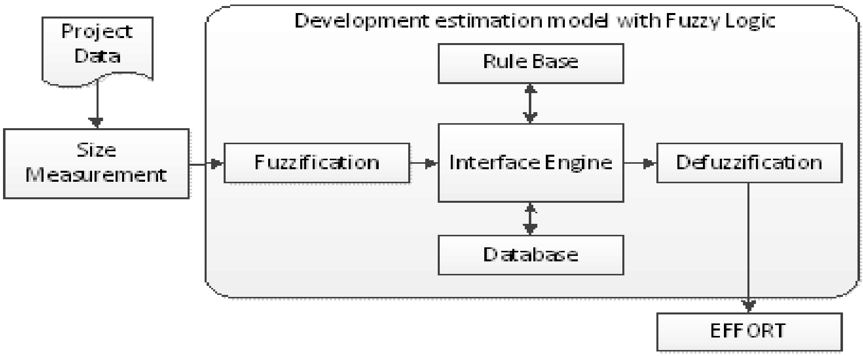


Figure 1. Proposed Effort Estimation Methods

Size measurement variables in this research is using the variable FHSWebEE method for estimating the functional size and hypermedia measurement by Rosmina and Suharjito [22] and the estimated size of the mobile project ever undertaken by Souza and Aquino Jr. There were three input variables to measure the mobile web effort estimation, namely Functional Size

Measurement, Hypermedia Size Measurement and Characteristics of Mobile Application Size Measurement as shown Figure 2.

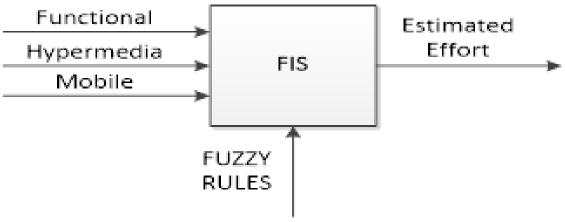


Figure 2. Fuzzy Effort Estimation Framework of Web-Based Mobile Apps

3.1. Functional Size Measurement

There are many functional size measurement methods, and one of those methods is OOmFPWeb by Abrahao. OOmFPWeb is a FSM method for object oriented system [4]. It is a combination of FSM method described in IFPUG meta model (ISO/IEC, 2003a) and OOWS method. This method can measure functional size of a web application from the user requirement specification. There are five elements that need to be counted in OOmFPWeb [23]:

a. Internal Logical Files (ILF): It is a class that encapsulates a set of data items (attributes) representing the state of the objects in each class.
b. External Interface Files (EIF): It is a legacy view that is defined as a filter placed on a class by a preexisting system.
c. External Input (EI): It is a service defined in a class or legacy view, since a service always change the state of the class (altering the behavior of the system).
d. External Inquiries (EQ): It is an Instance Interaction Unit (IIU), Population Interaction Unit (PIU), and Master Detail Interaction Unit (MDIU) defined in the Presentation Model. Their intent is to present information to user. The pattern must perform some calculations, or use some derived attribute.
e. External Output (EO): It is an Instance Interaction Unit (IIU), Population Interaction Unit (PIU), and Master Detail Interaction Unit (MDIU) defined in the Presentation Model. Their intent is to present information to user without altering the system behavior.

Figure 3 illustrates how the five elements of Function Point work. ILF and EIF are included in data function category. EI, EQ, and EO are included in transactional function category. Complexity of a data function depends on total Data Element Type (DET) and total Record Element Type (RET), while complexity of a transactional function depends on total Data Element Type (DETTransaction) and total File Type Referenced (FTR). There are measurement rules to count total DET and RET or DET and FTR for every function. Every different function type has different measurement rules. The measurement rules from OOmFPWeb can be seen in [23]. To estimate functional size of a project, we can use object diagram which can count data and transaction function easier.

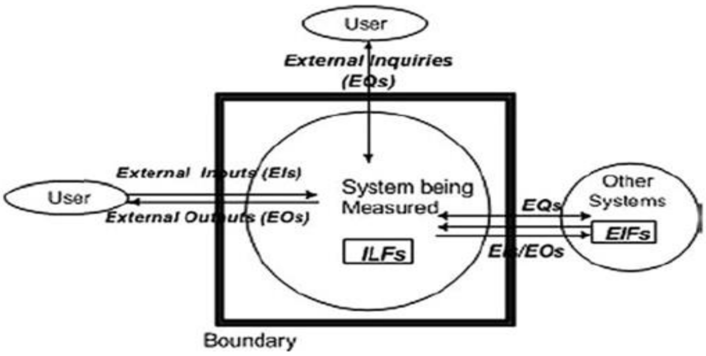


Figure 3. FPA View on Functional Size [23]

For every function found, a weight according is given to their complexity. Table 1 describes the weight for every function according to their type and complexity level provided in the IFPUG counting manual [16].

Table 1. Complexity Weight of Every Function Type

Function Type	Low	Average	High
ILF	7	10	15
EIF	3	4	6
EI	3	4	6
EO	4	5	7
EQ	3	4	6

Next, we can count total functional size or OOmFPWeb. OOmFPWeb is calculated as shown equations (1)-(3)

$$OOmFPWeb = OOmFPD + OOmFPT \quad (1)$$

where:

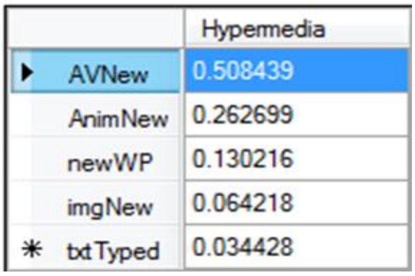
$$OOmFP_D = \sum_{i=1}^{n} OOmFP_{class(i)} + \sum_{j=1}^{m} OOmFP_{legacyview(j)} \quad (2)$$

$$OOmFP_T = \sum_{i=1}^{x} OOmFP_{services(i)} + \sum_{j=1}^{y} OOmFP_{IIU \mid PIU \mid MDIU(j)} \quad (3)$$

where, n is total of class defined in project to be measured, m is total of legacy view found in project, x is total services found in all classes, and y is total interface found in project.

3.2. Hypermedia Size Measurement

The hypermedia size measurement is calculated by used several factors from metrics of Mendes. The Mendes' factors what is used in this research were number of new video or audio, new animations, new web pages, new images, and typed text. By used Analytical Hierarchy Process (AHP) [24], the calculation of hypermedia is obtained. Figure 4 shows the hypermedia calculation using AHP.



		Hypermedia
▶	AVNew	0.508439
	AnimNew	0.262699
	newWP	0.130216
	imgNew	0.064218
✳	txtTyped	0.034428

Figure 4. The Weight of Hypermedia Factors

By weighting the results, it can be concluded as shown equation (4)

$$Hypermedia = 0.508439(AVNew) + 0.262699\ (AnimNew) + 0.130216\ (newWP) + 0.064218(imgNew) + 0.034428(txtTyped) \quad (4)$$

where AVNew is new video or audio, AnimNew is new animation, newWP is new web pages, imgNew is new images, txtTyped is text be typed.

3.3. Characteristics of Mobile Application Size Measurement

Souza and Aquino proposed method which can measure characteristics of mobile application from user requirement specifications [5]. Productivity factors:

Functionality requirements: compatibility with the needs of the end user, the complexity of the requirements.

a. (- -) Complex and critical application area (thousands of FPs), multiple users and multicultural system.
b. (-) Interoperable application area with some complex characteristics, requiring special understanding from users and developers.
c. (+ / -) Partly automated, integrated application area and a medium size application (between 600 and 1000 FPs) with standard security requirements.
d. (+) Application area mostly automated and application with less than 5 interfaces with other systems; there are specific security requirements. 58 Computer Science & Information Technology (CS & IT)
e. (+ +) Very mature application area, simple and easy, a small stand-alone application (less than 200 FPs) for a small group of users.

Reliability requirements: maturity, tolerance to faults and recovery for different types of use cases.

a. (- -) Malfunctions may put in danger human lives and cause significant economic or environmental losses.}
b. (-) The software is part of a large real-time system where all the failures of operation will cause problems to many other applications.}
c. (+ / -) Not more than 2 hours of downtime is acceptable, but the system recovery routines are appropriate.
d. (+) Need for non-continuous operation, but daily.
e. (+ +) Need for periodic operation. Pausing for a few days will not cause any damage to the organization.

Usability requirements: understandability and easiness to learn the user interface and workflow logic.

a. (- -) A large number of different types of end users around the world.
b. (-) 2 or 3 different types of users with different skills.
c. (+ / -) A large number of end users with equal abilities.□
d. (+) No more than tens or hundreds of homogeneous users in perhaps more than one location.
e. (+ +) Only a few users, all located on one site.

Efficiency requirements: effective use of resources and adequate performance in each use case and under a reasonable workload.

a. (- -) Complex database with millions of data records and transactions per day, thousands of simultaneous end users.
b. (-) Large database, hundreds of simultaneous end users, critical response most of the time.
c. (+ / -) Large database, less than millions of data records and less than hundreds of simultaneous end users.
d. (+) Medium database in volume and structure, simple and predictable data requests from some simultaneous end users.
e. (+ +) Simple and small database without simultaneous end users or complex data requests.

Maintainability requirements: lifetime of the application, criticality of fault diagnosis and test performance.

a. (- -) Very large strategic software (over 20 years of lifetime) in a volatile area of business, with frequent changes in laws, regulations and business rules.
b. (-) Large software (10-20 years of lifetime), and frequent changes in laws, regulations and business rules.
c. (+ / -) Medium size software (5-10 years of lifetime), monthly changes in laws, regulations and business rules.
d. (+) Small software, rarely changes (2 to 5 years of lifetime).
e. (+ +) Temporary software (less than 2 years of lifetime), without modifications.

Portability requirements: adaptability and instability to different environments, to the architecture and to structural components.

a. (- -) Software users are located in many types of organizations, with various platforms (hardware, browsers, operating systems, middleware, protocols, etc), various versions and various update frequencies.
b. (-) The software must operate on some different platforms (hardware, browsers, operating systems, middleware, protocols, etc) and in various versions of each of them.
c. (+ / -) Each version of the software must run on multiple versions of a given platform (hardware, browser, operating system, middleware, protocols, etc), and the frequencies of update of the users are quite predictable.
d. (+) The software must run on a given platform (hardware, browser, operating system, middleware, protocols, etc), but the use of system-level services is limited because the upgrade process is partial.
e. (+ +) Software must be run on a particular platform (hardware, browser, operating system, middleware, protocols, etc), but the upgrade process is completely controllable.

Performance Factors:

a. (-) The application should be concerned with the optimization of resources for a better efficiency and response time.
b. (+ / -) Resource optimization for better efficiency and response time may or may not exist.
c. (+) Resource optimization for better efficiency and response time should not be taken into consideration.

Power Factors:

a. (-) The application should be concerned with the optimization of resources for a lower battery consumption.
b. (+ / -) Resource optimization for lower battery consumption may or may not exist.
c. (+) Resource optimization for a lower battery consumption should not be taken into consideration.

Band Factors:

a. (-) The application shall require the maximum bandwidth.
b. (+ / -) The application shall require reasonable bandwidth.
c. (+) The application shall require a minimum bandwidth.

Connectivity Factors:

a. (-) The application must have the maximum willingness to use connections such as 3G, Wifi, Wireless, Bluetooth, Infrared and others.
b. (+ / -) The application must have reasonable predisposition to use connections such as 3G, Wi-Fi and Wireless.
c. (+) The application must have only a predisposition to use connections, which can be: 3G, Wi-fi, Wireless, Bluetooth, Infrared or others.

Context Factors:

a. (-) The application should work offline and synchronize.
b. (+ / -) The application should work offline and it is not necessary to synchronize.
c. (+) The application should not work offline.

Graphic Interface Factors:

a. (-) The application has limitations due to the screen size because it will be mainly used by cell phone users.
b. (+ / -) The application has reasonable limitation due to the screen size because it will be used both by cell phone and tablet users.
c. (+) The application has little limitation due to the screen size because it will be mainly used by tablet users.

Input Interface Factors:

a. (-) The application must have input interfaces for touch screen, voice, video, keyboard and others.
b. (+ / -) The application must have standard input interfaces for keyboard.
c. (+) The application must have any one of the types of interfaces, such as: touch screen, voice, video, keyboard or others.

Caption factors:

a. □“+ +” = [1.10] Excellent situation, much better circumstances than in the average case
b. □“+” = [1.05] Good situation, better circumstances than in the average case

c. □“+ / -” = [1.0] Normal situation
d. □“-” = [0.95] Bad situation, worse circumstances than in the average case
e. □“- -” = [0.90] Very bad situation, much worse circumstances than in the average case

By requirements and factors the results, it can be calculated as shown equation (5) and (6).

$$Mobile\ Application\ Characteristics = \\ -\frac{Productivity+Performance+Power+Band+Connectivity+Contex+Graphic+Input}{8} \quad (5)$$

where:

$$Productivity = \frac{Functionality+Reliability+Usability+Efficiency+Maintainability+Portability}{6} \quad (6)$$

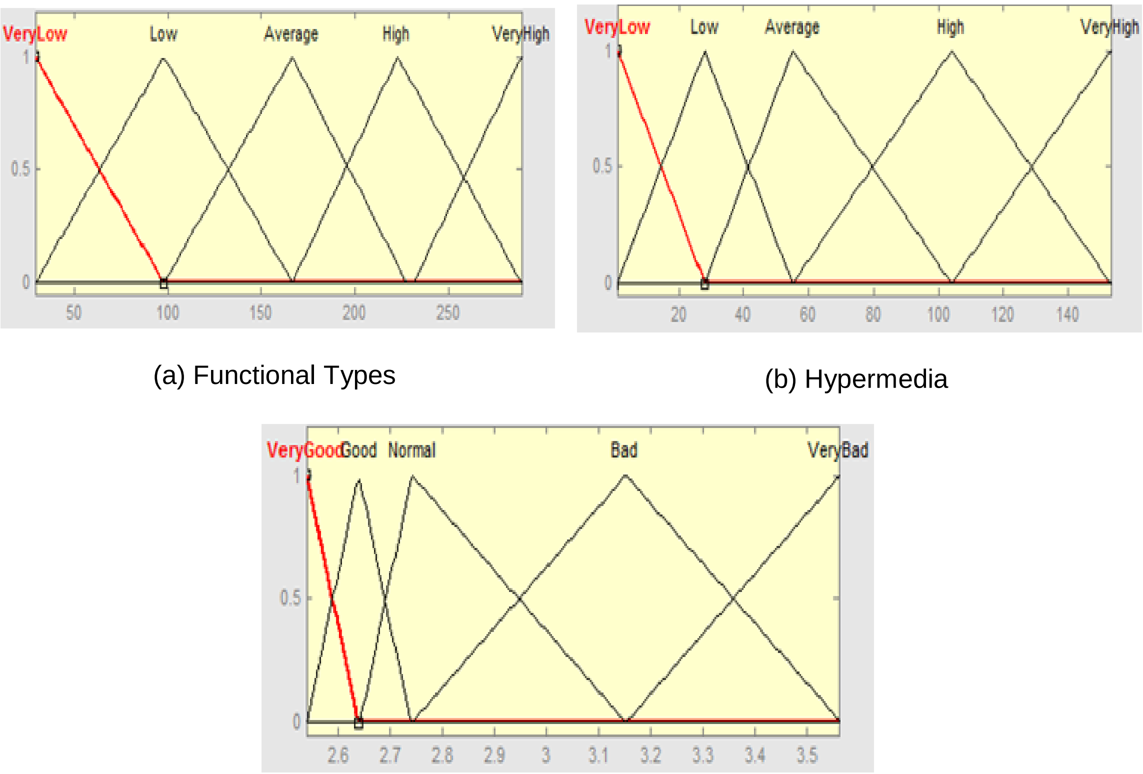
Due to Souza made application easier weighting the higher weight, then made the conversion to make it easier to apply to the fuzzy logic. The proposed conversion by us is shown Table 2.

Table 2. Weighted Conversion Mobile Application Characteristics

Souza & Aquino					Converted Result				
Very Good	Good	Normal	Bad	Very Bad	Very Good	Good	Normal	Bad	Very Bad
1.1	1.05	1	0.95	0.9	1	2	3	4	5

3.4. Fuzzy Logic

The functional, hypermedia and characteristics of mobile applications are used as input variables to the membership function and contains 125 rules as shown Figure 5 and Figure 6.



(a) Functional Types

(b) Hypermedia

(c) Characteristics of Mobile Application

Figure 5. Fuzzy Membership Function of Input Variables

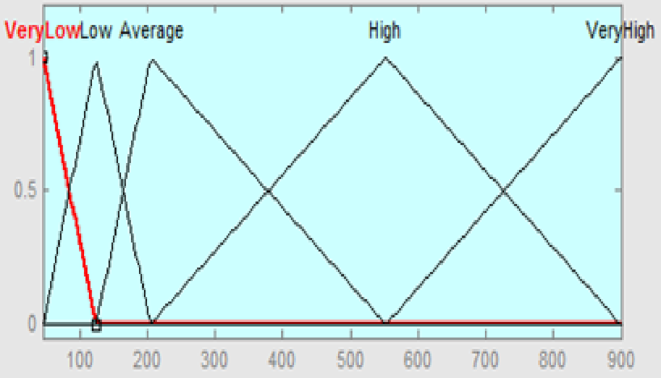


Figure 6. Fuzzy Membership Function of Output Variables

4. Results and Analysis

This research uses 30 web-based mobile application projects from several software houses in Indonesia to evaluate the proposed model. The thirty projects are evaluated by counting total function point, total hypermedia component and characteristic mobile used. The estimation's accuracy of the proposed model is checked by omitting a group of projects (30 projects) as dataset. The accuracy prediction of this model is calculated used MMRE, MdMRE, and Pred(n). Figure 7 denotes the comparison between the actual effort of the project estimated, OOmFPWeb and the result of effort estimation with the proposed method.

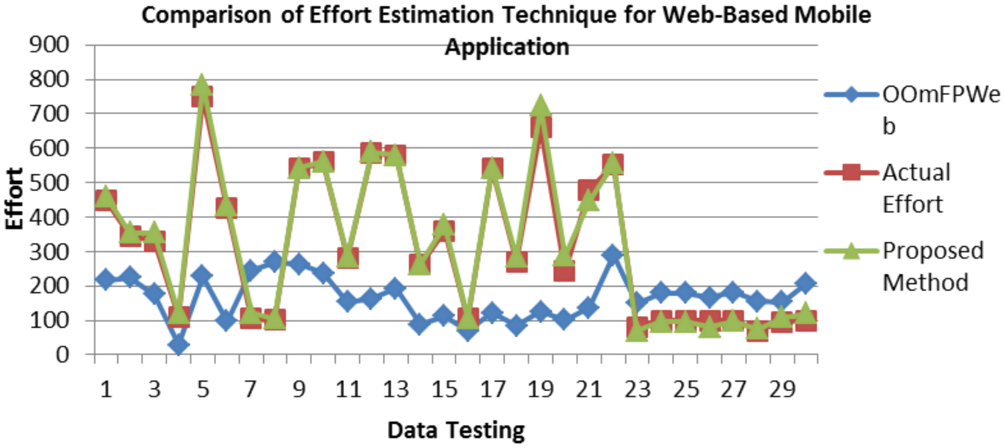


Figure 7. Estimation result between OOmFPWeb and Proposed Method

Table 3 shows the result of acuracy prediction. The result shows, that the proposed method is better than just using function point (OOmFPWeb) as variable for effort estimation. The mean of relative error of the proposed method is about 5.9%, the median of relative error of proposed method is about 4.5% and 100% from the tested projects have error level below of equals to 25%. The mean of relative error of OOmFPWeb is about 7.3%, median of relative error is about 70%, and 0% of the tested projects have error level below or equals to 25%.

Table 3. Accuracy Prediction Result

Accuracy Measurement	Proposed Method	OOmFPWeb
MMRE	0.059215	0.733386
MdMRE	0.044722	0.693111
Pred(25)	1	0

5. Conclusion

This paper proposes a model to estimate effort needed by an object oriented web-based mobile application project development. The procedure of this model is to estimate functional size measurement using OOmFPWeb, total hypermedia size and the characteristics of mobile application in a web-based mobile application. Total hypermedia size is retrieved by using several variables from web metrics of Rosmina and using AHP and total characteristic of mobile application by Gibeon. After functional, hypermedia and characteristic mobile size retrieved, the project data size is compared with the actual effort with Mamdani fuzzy logic. The evaluation is done for this model by using 30 project gather from several web mobile companies. The evaluation results show that the proposed method is better than just using OOmFPWeb to estimated effort needed.

References

[1] DM Rao. Mobile Southeast Asia Report. Mobile Monday. 2012.

[2] K Schwalbe. Information Technology: Project Management. 6th Edition ed., C Learning, Ed., Boston. Course Technology. 2010: 8-9.

[3] E Mendes, N Mosley, S Counsell. *Investigating Early Web Size Measures for Web Cost Estimation.* Procedings of the EASE' 2003 Conference. 2003: 1-22.

[4] S Abrahao, G Poels. A Family of Experiment to Evaluate a Fuctional Size Measurement Procedure for Web Applications. *Journal of Systems and Software.* 2009; 82(2): 253-269.

[5] LS Souza, GS Aquino Jr. Estimating the Effort of Mobile Application Development. CS & IT. 2014: 45-63.

[6] M Shepperd, C Schofield, B Kitchenham. *Effort Estimation Using Analogy.* Proceeding of ICSE-18. 1996.

[7] I Myrtveit, E Stensrud. A Controller Experiment to Assess the Benefits of Estimating with Analogy and Regression Model. *IEEE Transaction on Software Engineering*. 1999; 25(4): 510-525.

[8] MJ Shepperd, C Schofield. Estimating Software Project Effort Using Analogies. *IEEE Transaction on Software Engineering.* 1997; 23(11): 736-743.

[9] S Suharjito, J Jimmy, AS Girsang. Mobile Decision Support System to Determine Toddler's Nutrition Using Fuzzy Sugeno. *International Journal of Electrical and Computer Engineering (IJECE).* 2017; 7(6): 3683-3691.

[10] AJ Lubis, E Susanto, U Sunarya. Implementation of Maximum Power Point Tracking on Photovoltaic Using Fuzzy Logic Algorithm. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 2015; 13(1): 32-40.

[11] V Sharma, HK Verma. Optimized Fuzzy Logic Based Framework for Effort Estimation in Software Development. *IJCSI International Journal of Computer Science.* 2010; 7(2): 30-38.

[12] NVRS Kumar, A Mandala, MV Chaitanya, G Prasad. Fuzzy Logic for Software Effort Estimation Using Polynomial Regression as Firing Interval. *IJCTA.* 2011: 1843-1847.

[13] PR Prasad, KR Sudha, PS Rama. Application of Fuzzy Logic Approach to Software Effort Estimation. *IJACSA International Journal of Advance Computer Science and Application.* 2011; 2(5): 87-92.

[14] S Malathi, DS Sridhar. Optimization of Fuzzy Analogy in Software Cost Estimation Using Lingustic Variables. *Procedia Engineering.* 2012; 32: 177-190.

[15] Ziauddin, S Kamal, S Khan, JA Nasir. A Fuzzy Logic Based Software Cost Estimation Model. *International Journal of Software Engineering and Its Application*. 2013; 7: 7-18.

[16] Suharjito, S Nanda, B Soewito. *Modeling software effort estimation using hybrid PSO-ANFIS. Int. Semin. Intell. Technol. Its Appl*. 2016: 219–224.

[17] ND Gracia, CM Lopez, A Chavoya. A Comparative Study of Two Fuzzy Logic Model for Software Development Effort Estimation. Procedia Technology. Mexico. 2013.

[18] A Cowderoy, A Donaldson, J Jenkins. *A Metrics Framework for Multimedia Creation.* Proc. 5th IEEE International Journal Software Metrics Symposium. 1998.

[19] E Mendes, S Counsell. *Web Development Effort Estimation using Analogy.* Proc. 2000 Australian Software. 2000: 203-212.

[20] D Reifer. Web Development: Estimating Quick-to-Market Software. *IEEE Software.* 2000: 57-64.

[21] D Cleary. *Web-based development and functional size measurement.* Proceedings of IFPUG Annual Conference, San Diego, USA. 2000.

[22] Rosmina, Suharjito. FHSWebEE: An Effort Estimation Model for Web Application. *Procedia Engineering.* 2012; 50: 613-622.

[23] S Abrahao, G Poels, O Pastor. A Functional Size Measurement Model for Object-Object Conceptual Schemas: Desain and Evaluation Issue. *Software & System Modelling.* 2006; 5(1): 48-71.

[24] RW Saaty. The Analytic Hierarchy Process-What It Is and How It Used. *Journal of Mathematical Modelling.* 1987; 9(3-5): 161-176.