

AN EMPLOYING A MULTISTAGE FUZZY ARCHITECTURE FOR USABILITY OF OPEN SOURCE SOFTWARE

Roba Al-Soub

Mutah University, Information Technology Department, Jordan

ABSTRACT

The Demand for Open Source Software (OSS) is increasing day by day. However, its end users still face challenges using such software. Therefore, this study is conducted to propose a fuzzy usability model to be an approach for evaluating the usability of the open source software. In order to propose such model, six common usability characteristics have been considered, namely: Learnability, understandability, attractiveness, operability, efficiency, and memorability. Some of these characteristics are related to the open source software's features (Learnability, usability, attractiveness, operability); and the rest are related to the end users. Finally, the Matalab Simulink software (Fuzzy Logic Toolbox) has been employed to simulate and to validate the proposed approach model.

KEYWORDS

Open source software usability, usability characteristics, usability fuzzy model, and open source software usability model.

1. INTRODUCTION

These days, Open Source Software (OSS) projects are acquiring much popularity over development methodologies of commercial and proprietary software, due its ability to enable academics, researchers, and programmers to share their programming experience [1]. A lot of open source software have successfully been developed and deployed to users. Such these software is a Mozilla web browser, OpenOffice, Linux Operating System, etc. [2]. However, users of OOS still encounter challenges using the software interface. In this respect, end users of many popular open source software share comments that blend praise of its functionality with criticism of its usability [3]. It could be said that, poor usability hurts the users, damage developer's reputation, and increase distrust of software. Thus, open source software developers, should not only focus on the functionality, but also on the OSS usability.

In the context of software quality, International Organization for Standardization and International Electro Technical Commission, ISO/IEC 9126-1 define usability as the capability of a software to be understood, learned, used, and attractive to the end user, when used under particular conditions [5]. Here, in order to evaluate software usability, it is divided into understandability, learnability, operability and attractiveness. Additionally, although several studies provide more valuable characteristics for evaluating software usability for instance Efficiency, and memorability [4],[6],[7], and [8], there is no exact model for evaluating usability specially the usability of OOS, due its fuzzy characteristics, where, the usability model was defined as a conceptual view that explain usability factors for a specific software [8]. Consequently, this research has been done to propose a fuzzy approach model to evaluate the usability of OSS including factors should be considered when developing an OSS.

2. CHARACTERISTICS AFFECTING OSS USABILITY

For measuring the open source software usability six factors should be taken into consideration. These characteristics provide a rich conceptual framework for integrating previous research findings and adapting features of open source software.

2.1. Learnability:

It is related to the capability of the open source software to be easily learned by the end users [5].

2.2. Understandability

It refers to the capability of the open source software to be understood by the end user, whether the software are suitable and how can be used for particular task [6].

2.3. Operability

It is the capability of the open source software to enable the end user to operate and control it [9]. The operability metrics should measure whether a user operate and control the software.

2.4. Attractiveness:

This usability characteristic refers to the capability of the open source software to be attractive to its users [6]

2.5. Efficiency:

It is directly related to the number of tasks a user of the open source software can perform in a specific period of time [6].

2.6. Memorability:

It is related to the capability of the end user to return to the open source software's previous state without starting away from the beginning [7].

3. FUZZY REASONING APPROACH MODEL

3.1. Traditional Fuzzy Model of Open Source Software Usability

As previously mentioned, there is no exact usability model to evaluate software usability in general and usability of open source software in more specific. Furthermore, although many methods have been used to evaluate software usability there is no exact approach among them. Thus, this study has been conducted to propose a new fuzzy approach model to evaluate the open source software usability.

The Fuzzy logic has gained much more attention by scholars as it considers the fuzzy value instead the binary values. Further, the fuzzy logic model could be created even with little or no data [10]. This paper has proposed a fuzzy approach model to measure usability of the open source system. The model involves six fuzzy usability characteristics including learnability,

understandability, efficiency, memorability, operability, and attractiveness. Figure 1 typically shows the block diagram of open source software usability fuzzy model.

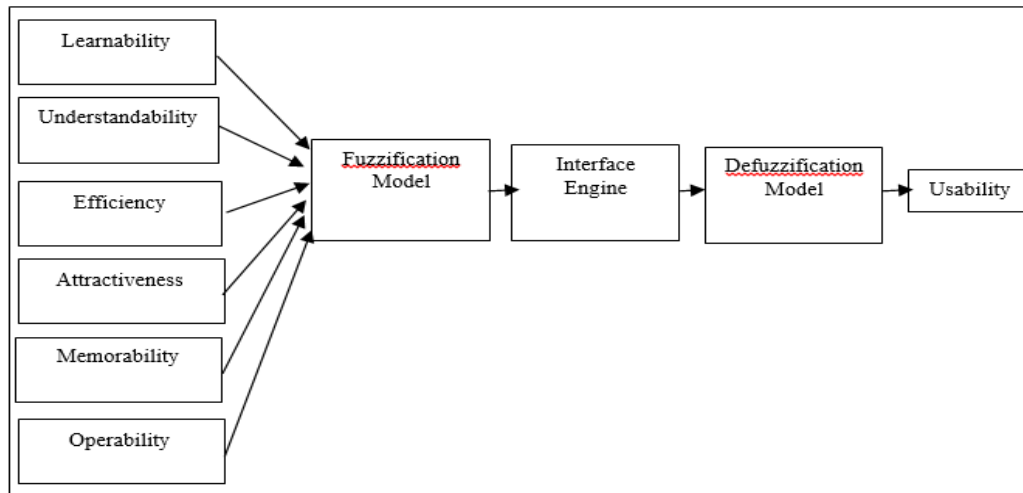


Fig. 1 Traditional Fuzzy Model for usability of open source software

Figure 1 considers all the inputs (the usability characteristics) together, so that will lead generating too many rules. Additionally, it is difficult for the experts consider all formulates rules with proper emphasis. Since each input parameter has three linguistic values (Low, Moderate and Strong). Hence, the open source software with six usability characteristics will have a maximum number of $3^6 = 729$ rules. This means, the Matlab-Fuzzy Tool Box isn't applicable, since the number of inputs is limited to two in the Matlab [13].

The solution lies in the Stage Wise Fuzzy Reasoning Approach, which reduces the number of rules by dividing the whole system into many fuzzy interface stages [11] and [12] and which effectively evaluate usability of any open source software. This solution makes it possible to use the Stage wise Fuzzy Reasoning, since the number of inputs is limited to two in each stage [13].

3.2. Stage Wise Fuzzy Reasoning

If the usability characteristics which involved in the proposed model are considered, it could be seen that the usability characteristics such as learnability, understandability, operability, and attractiveness are related to the ability of the software to inspire the user to perform the task correctly. On the other hand, efficiency and memorability characteristics reflect the end user's ability to perform the task. With this understanding, the usability characteristics in the proposed model can be grouped as in Table 1.

Table 1: Grouping of Elements

Group	Critical Element
Software Related	Learnability, Understandability, Operability and Attractiveness
User Related	Efficiency and Memorability

3.3 Multi Stage Fuzzy Logic Model for Usability of Open Source Software

In order to design multistage Fuzzy model for usability of open source software three steps would be conducted.

Step 1: defining Fuzzy Interface System (FIS):

Total five Fuzzy Interface System (FIS) namely soft1, soft2, software-Interm, user, and software-f have been created in Matlab using a Fuzzy Logic toolbox [13]. Consequently, input/output variables, their membership functions, and fuzzy control rules have also been created for each FIS. Figure 2 shows an example of FIS.

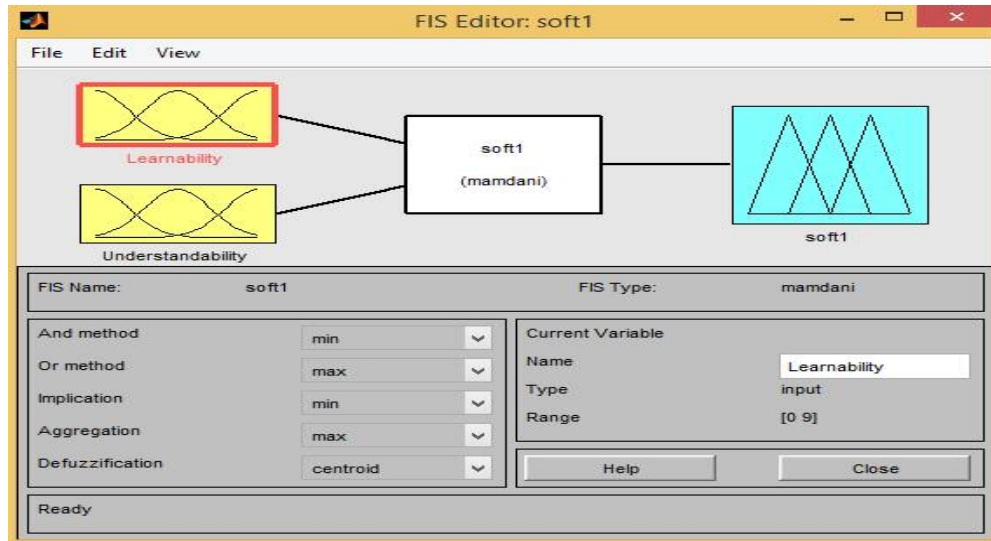


Figure 2. FIS of Learnability and Understandability

Step 2: defining universe of Discourse and fuzzification of usability characteristics:

Each of the six usability characteristics have been given a universe of discourse (UOD) of [0-9]; and have been fuzzified with three linguistic values (fuzzy sub set: Low, Moderate, and Strong) using linear triangular membership functions [14]. On the other hand, in order to achieve more accurate output, all the fuzzified output parameters have been fuzzified with four linguistic values (fuzzy sub set: Low, Moderate, Strong, and Extreme). Figure 32 shows three examples of fuzzified input and output parameters.

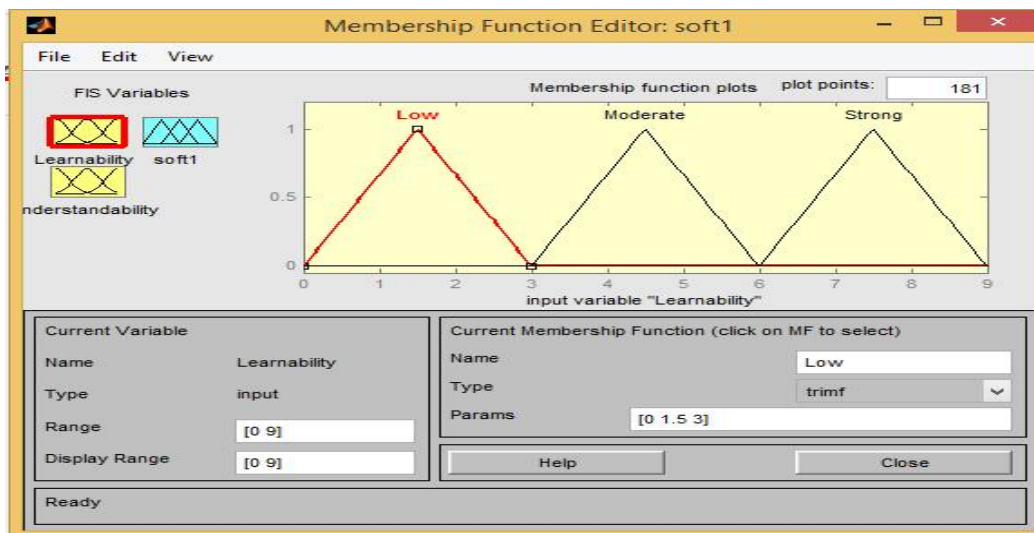


Figure 3a. Fuzzified Input Parameter- Learnability

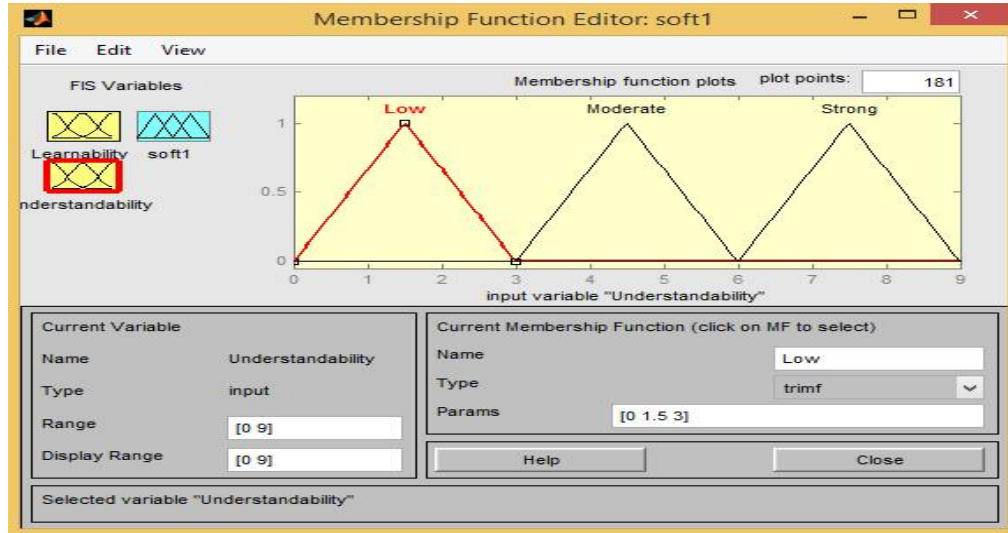


Figure 3b. Fuzzified Input Parameter- Understandability

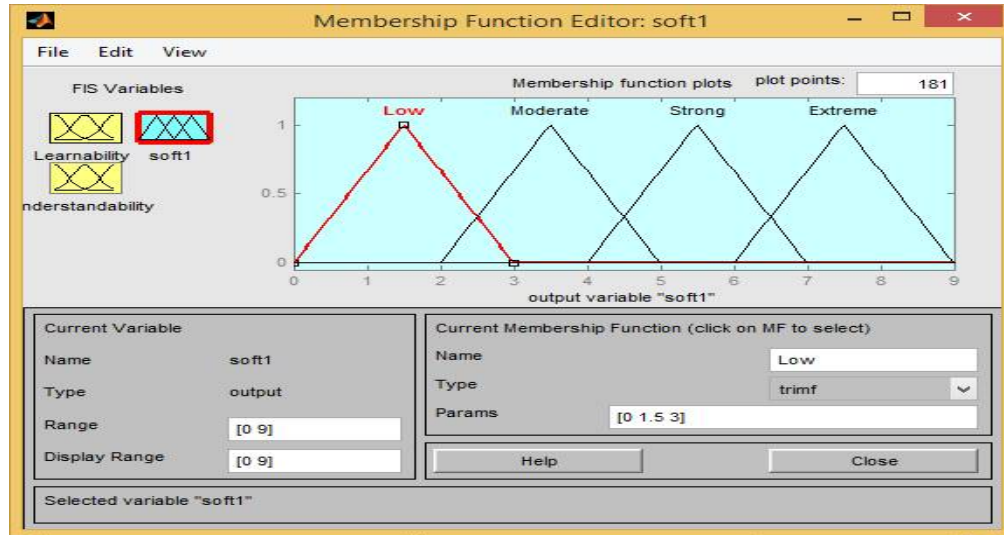


Figure 3c. Fuzzified Output Parameter-soft1

Step 3: defining fuzzy control rules:

The fuzzy control rules of first fuzzy interface system FIS (soft1) have been defined as follows: learnability and understandability are the fuzzy input variables and soft1 is the fuzzy output variable. Fuzzy linguistic values of input/output variables are set as [Low, Moderate, Strong, or Extreme]. The fuzzy control rules have been defined based on the opinions of academicians, researchers, and open source software developers. Table 2 shows the fuzzy control rules of FIS1.

Table 2. Rules of First FIS (soft1)

Num	Learnability	Understandability	Usability of Soft1
1	L	L	L
2	L	M	L
3	L	S	M

4	M	L	L
5	M	M	M
6	M	S	S
7	S	L	M
8	S	M	S
9	S	S	S

As seen from the above table, this FIS includes total 9 rules. For all combinations usability is either classified as Low, Moderate, or Strong. The rules of soft2 and user FIS had been defined in the same way.

Regarding, soft-interm FIS that is an intermediate stages fuzzy controller, output variables (soft1 and soft2) of soft1 and soft2 FIS respectively, are taken as input variables and soft-interm is an output variable. The fuzzy linguistic values of output variables are again at [Low, Moderate, Strong, and Extreme]. Table 3 shows fuzzy control rules of this FIS. These rules are also defined based on the opinion of academicians, researchers, and open source software developers.

Table 3. Rules of soft-interm FIS

Num	Soft1	Soft2	Usability of soft-interm
1	L	L	L
2	L	M	L
3	L	S	M
4	L	E	S
5	M	L	L
6	M	M	M
7	M	S	S
8	M	E	S
9	S	L	M
10	S	M	S
11	S	S	S
12	S	E	E
13	E	L	M
14	E	M	S
15	E	S	E
16	E	E	E

Software-f FIS is the final stage fuzzy controller. The output variables of user and soft-interm FIS are taken as an input variables and software-f is an output variable. Fuzzy linguistic values of output variables are at [Low, Moderae, Strong, and Extreme]. Table 4 shows fuzzy control rules. These rules are also defined based on the experience of academicians, researchers, and open source software developers.

Table 4. Rules of Final FIS (software-f)

Num	software-u	user-u	Usability of OOS
1	L	L	L
2	L	M	L
3	L	S	M
4	L	E	S
5	M	L	L
6	M	M	M

7	M	S	S
8	M	E	S
9	S	L	M
10	S	M	S
11	S	S	S
12	S	E	E
13	E	L	M
14	E	M	S
15	E	S	E
16	E	E	E

3.3 Simulation Of Multi Stage Fuzzy Model For Usability Of OSS

In this step, Matalab Simulink software (Fuzzy Logic Toolbox) has been employed, in order to develop an approach model for proposed stage wise fuzzy reasoning. This model connects soft1, soft2, user, soft-Interm, and software-f. Fig.4 shows a fully functional the fuzzy logic model.

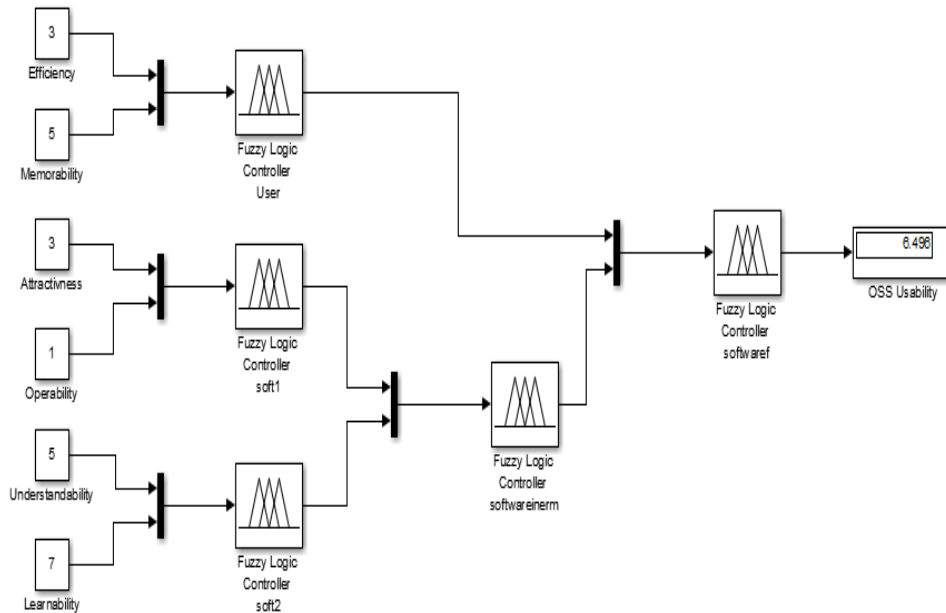


Figure 4. Fuzzy Approach Model of OOS Usability

Tables 5 and 6 present the open source software usability depending on various simulations for different sets of critical elements rating.

Table 5. Simulation Data Set 1

Critical element	Rating	Stage wise				OOS Usability
		Soft1	Soft2	User	Soft-interm	
Learnability	5	5.5				
Understandability	7					

Operability	1				3.5
Attractiveness	3		1.5		3.5
Efficiency	3				
Memorability	5	3.5			

The data in table 5 reveal that traditional non-fuzzy approach rates the software-related characteristics to 3.5 (moderate) and user-related characteristics to 3.5 (moderate). Whereas, the result of fuzzy reasoning approach clearly shows that the overall rating of the software usability is 3.5 (moderate). However, consider the scenario in table 6, where the understandability is rated to 1. Here, traditional non-fuzzy approach would rate software-related characteristics to 1.5 (low). Whereas, fuzzy reasoning approach would rate the overall usability of the software to 1.5 (low).

Table 6. Simulation Data Set 2

Critical element	Rating	Stage wise				OOS Usability
		Soft1	Soft2	User	Soft-interm	
Learnability	5					1.5
Understandability	1	1.5				
Operability	1					
Attractiveness	3		1.5		1.5	
Efficiency	3					
Memorability	5	3.5				

4. CONCLUSION

This study has been conducted in order to propose a comprehensive fuzzy approach model to measure the usability of open source software. The inputs of this proposed model consists of six usability characteristics including: learnability, understandability, operability, attractiveness, efficiency, and memorability. These characteristics have also been classified with their relevance. In order to reduce the number of used rules, the fuzzy logic model has been used in multi stages. Looking at the simulation data, the stage-wise fuzzy reasoning has a more logical approach to usability analysis. Furthermore, the OSS's developers and organizations have the flexibility to give different weights to different usability characteristics as per their requirements.

REFERENCES

- [1] Wahyu A. R.E., Wardoyo R., and Eko J.I. (2011) "Modularity Index Metrics for Java-Based Open Source Software Projects". *(IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 2, No. 11.*
- [2] Hall, James. (2014). Usability Themes in Open Source Software. Retrieved from the University of Minnesota Digital Conservancy, <http://hdl.handle.net/11299/163071>.
- [3] Aruna M., M.P. Suguna Devi M.P, Deepa M. (2008) "Measuring the quality of software modularization using coupling-based structural metrics for an OSS system", Proceeding of the First International Conference on Emerging Trends in Engineering and Technology.

- [4] Raza, A., Capretz, L. F., & Ahmed, F. (2011) “ An empirical study of open source software usability: The industrial perspective.”, International Journal of Open Source Software and Processes (IJOSSP), 3 (1), 1-16.
- [5] International Organization for Standardization. ISO/IEC 9126-1: Software engineering –product quality – Part 1: Quality model (1st ed., pp.9–10). Geneva, Switzerland: ISO/IEC.
- [6] ISO 9241. “ Ergonomics requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability”.
- [7] Nielsen J. and V. Philips. (1993) “Estimating the relative usability of two interfaces: Heuristic, formal, and empirical methods compared,” in Proc. ACM/IFIP Human Factors Computing Systems (INTERCHI), Amsterdam, The Netherlands, pp. 214–221
- [8] Madan, A., & Dubey, S. K. (2012) “ Usability evaluation methods: a literature review”. International Journal of Engineering Science and Technology, 4(2), 590-599.
- [9] Bertoa, M., & Vallecillo, A. “ Usability metrics for software components”. In 8th International Workshop on Quantitative Approaches in Object-Oriented Software Engineering (QAOOSE'2004), Oslo, Norway.
- [10] Shaout, A., & Trivedi, J. (2013) “ Performance Appraisal System Using a Multistage Fuzzy Architecture.” International Journal of Computer and Information Technology (ISSN: 2279–0764) Volume.
- [11] Dahal, K., Hussain, Z. and Hossain, M.A. (2005) “Loan Risk Analyzer based on Fuzzy logic”, e-Technology, e-Commerce and e-Service, 2005. IEEE '05 proceedings. The 2005 IEEE International Conference on Digital Object Identifier, Page(s): 363 – 366.
- [12] Adnan Shaout and Minwir Al-Shammari, (1998) “Fuzzy logic modeling for performance appraisal systems A framework for empirical evaluation”, Journal of Expert Systems with Application, issue 14 pages 323-328.
- [13] MathWorks, Inc, & Wang, W. C. (1998). “ Fuzzy Logic Toolbox: for Use with MATLAB: User's Guide. MathWorks”, Incorporated.
- [14] John Yen, Reza Langari, (1999) “Fuzz Logic: Intelligence, Control and Information”, Prentice Hall - ISBN 978-81-317-0534-6.