International Journal of Science and Research (IJSR)
ISSN: 2319-7064
ResearchGate Impact Factor (2018): 0.28 | SJIF (2018): 7.426

The Social Software Learnability Prediction (SSLP) Tool

Masese. B. Nelson

Department of Computer Science, Kabarak University, Kenya

Abstract: ***There are many social software's in the market, but not all of them are being utilized probably due to complicated user interface features, memorability of the software is difficult or due to very difficult language used in the application. The main purpose of this paper is to design the mobile social software learnability prediction tool, to assist the prediction of the learnability of the social software before it is released to the market. A sample of 361 respondents was selected, with 345 respondents returning feedback. Primary data was collected through the use of questionnaires targeting mobile social users in Central Rift valley of Kenya. Social networks were targeted include, WhatsApp, Facebook and Twitter. Data analysis was done using descriptive statistics. Principal component analysis was used to selected the most significant variables that were uses in the design of the tool, the variables that were considered include Customization, Satisfaction, Software consistency, User interface features, Language complexity and Memorability. Results from the tool were used in the prediction of social software learnability.***

Keywords: Principal component analysis, User interface features, Language complexity, Memorability

1. Introduction

Learnability is a quality of products and interfaces that allows users to quickly become familiar with them and able to make good use of all their features and capabilities. Learnability is one component of usability and is often heard in the context of user interface or user experience design, as well as usability and user acceptance testing [1].

A very learnable interface or product is sometimes said to be intuitive because the user can immediately grasp how to interact with the system. First-time learnability refers to the degree of ease with which a user can learn a newly-encountered system without referring to documentation, such as manuals, user guides or frequently-asked questions lists [2]. One element of first-time learnability is discoverability, which is the degree of ease with which the user can find all the elements and features of a new system when they first encounter it. Learnability over time, on the other hand, is the capacity of a user to gain expertise in working with a given system through repeated interaction [3].

Learnability can be optimized by creating simple user interface designs that are predictable in layout and navigation. A great way to improve learnability is by finding out what the expectations of your users are before they use the application [4]. This can be in a focus group or through requirements engineering. Usability testing can also be used to discover how quickly end users pick up how to use the application. If the usability test is carried out several times with the same user at different intervals, you can also see how memorable the application was. Feedback can then be used to optimize the learnability of an application, increasing its chances of success on the market [5].

There are many social software's in the market, but not all of them are being utilized, is it a learnability issue? The main purpose of this paper is to design the mobile social software learnability prediction tool

2. Related works

The speed of adoption is not the only criteria why learnability matters. Application that looks familiar and provides an understandable interface will result in a lower bounce rate. This is especially useful for applications that try to boost their conversion rate [6]. A complex design scares users and they will resort to other tools that provide a clear interface. In the end, the goal of every application is to convert an occasional user into a repeated user and engage the user for interaction [7].

The learnability of application can be measured in the following ways:

a) **Effectiveness:** The number of functions learned, or the percentage of users who successfully learn and use the product [8].
b) **Efficiency:** The time it takes someone to learn (or re-learn) how to use a product, and their efficiency in doing so [9].
c) **Satisfaction:** The perceived value the person associates with their investment (time, effort, cost) in learning how to use the product [10].
d) **Errors:** The number of errors made, the ability to recover from those errors and the time it takes to do so[11]

2.1 Requirements for the SSLP Tool

The design of this tool was guided by factors affecting learnability of social software applications. To start with the need for such a tool was informed by confirming the concerns that the learnability of mobile social software that include memorability, user interface features and program complexity so as users can easily learn and utilize mobile social software as some software's are difficult to utilize and even for the user to explore all functionalities in the software system at any given time [12].

Principal component analysis was used to select the variables to be used in the design of the model. Principal components analysis is statistical technique used for data reduction. This method is applied to a single set of variables when the researcher is interested in discovering which variables in the set form coherent subsets that are relatively independent of one another. Variables that are correlated with one another but are largely independent of other sets of variables are combined into factors [13].

These factors allowed researcher to condense the number of variables in the analysis by combining several variables into one factor. The specific goals of principal components analysis is to summarize patterns of correlations among observed variables, to reduce a large number of observed variables to a smaller number of factors, to provide a regression equation for an underlying process by using observed variables.

The variables considered included customization (-0.42) satisfaction (-0.026), Consistency (-0.08), user interface features (0.013) program complexity (0.264), memorability(1.199) after the analysis it was found that satisfaction, customization, and software consistency had a non-significant negative regression weights of -0.026,-0.042,-0.076 and -0.008 respectively indicating respondents with higher scores on this scale were expected to have lower influence on learnability, after controlling for the intervening variables in the model this led to the exclusion of the variables in the model and it produced an r^2 of 0.8% which is very low. The remaining factors include user interface, program complexity and memorability; they are discussed below in detail [12].

The user interface analysis findings points out that interface features may affect learnability across the three social networks, the coefficients was found at 0.013. This implies that the type of user interface of particular software influences its learnability by either improving it or decreasing it when the interface is quiet unfriendly. A software should promote users ease of use also the results indicate that as the interface allow users to recover from errors, icons or commands becoming clear and likeable to the specific functions, provides the user with enough suggestions and prompt towards the right usage, provides the feedback when errors occur and provide the user with help facilities, all the factors discussed increase the tendency of learnability [12].

Program complexity is another factor that affects learnability findings had indicated that there is a direct relationship with learnability with the coefficients found at 0.264. The Analyzed data supported the relationship between learnability and program complexity. This implies that as the as the program in the social software and in the users' guide becomes clear and easy to understand learnability increases in equal measure. This could be further illustrated when the program is consistent for various operations and when the program is able to manage errors during data entry the learnability of the software is enhanced [12].

Another critical factor software memorability influences the learnability as the software's that are easy to remember are regarded to be learnable compared to software's that are hard to memorialize. The findings indicated that learnability will increase in a user when he or she is capable of remembering how to operate the icons earlier encountered, it had a coefficients of 1.199. Memorability as a factor influencing learnability as users having difficulty in executing such operations will have low learnability scores. Furthermore the finding implies that learnability will increase in a user when he or she is capable of remembering how to operate the icons earlier encountered. The regression coefficients obtained from the analysis formed the coefficients for the model [12].

2.2 Derivation of Metrics Based on the Goal Question Metric (GQM) Approach

After identifying the key variables in the model architecture the next step was to develop a framework that would help in identifying metrics that would be used to implement the proposed model. To do this the researcher used the goal question metric (GQM) approach for deriving metrics as proposed by [14].

The Goal-Question-Metric (GQM) approach is a proven method for driving goal-oriented measures throughout a software organization. With GQM, we start by defining the goals we are trying to achieve, then clarifying the questions we are trying to answer with the data we collect. By mapping business outcomes and goals to data-driven metrics, we can form a holistic picture of the Agile environment and clearly articulate how we are doing across the span of the design[15]. The following in order to ascertain the necessary metrics for the SSLP.

2.2.1 Deriving Metric for Memorability

Goal to improve learnability of mobile social software (reduce learnability challenge)
Question 1.1 How can memorability of mobile social index of a user be (MI) determined?
Metric1 1.1 Percentage of tasks executed for a over a specified time.

2.2.2 Deriving Metric for User Based Features

Question 1.2 How can user interface features affect mobile social user during the execution of different tasks (User interface index) be determined?
Metric 1.2.1 Percentage of tasks executed through user interface features over a specific time.

2.2.3 Deriving Metric for Program Complexity

Question 1.3 How can program complexity index (PCI) be determine?
Metric 1.3.1 Percentage of commands executed over a specified time.
The working formula to develop the model is as follows:
Learnability = 4.149+ (User interface features x 0.013) + (Program complexity x 0.264) + (Memorabilityx1.199) + 1.763

Where : Learnability is a constant holding a value of 4.149, while on the user interface, memorability and program complexity have constants as shown on the formula above but the user will enter values from for the respective sub metrics it will depend on the values that will be selected by the user where by the minimum value will be one and the maximum value will be five in the given question, after the user has selected the options they will be computed and the total will be achieved for each sub metric.

The minimum value will be seven when the user has selected the very lowest option only, for all the questions in the given sub metric and maximum value for each sub metric will be thirty five when the user has selected the highest option then it will be multiplied with the respective constants in each sub metric as stated in the above formula.

For the ideal situation of learnability the score will be above 64.25 percent meaning the user is confident of the user interface used in that particular software ,the user has good memorability of the commands used in that software and lastly the user can easy recall number commands used in the language and lastly he can execute and number of tasks even if some will be challenging to execute t any given time it cuts across the three social software's that were used in this research and will be considered as ideal situation for this model.

According to this model the best case scenario for learnability is indicated when a given user can execute all tasks which has been assigned or tasks on the particular software then his learnability scale will be ranked as the excellent. In this case after adding all sub metrics and applying the learnability formula, value of 35 was achieved when the user has selected excellent option which has five points multiplied by the seven variables used in the model, if the points earned are between 82.12 to 100 then it translates to best case the calculation are demonstrated below:

Learnability = 4.149+ ((User interface features) 35 x 0.013) + ((Program complexity) 35 x 0.264) + ((Memorability) 35 x 1.199) + 1.763 = 100%

These calculations will translate to 100 percent learnability for the given user, but depending on the selection for different sub metrics it must be between 82.12 to 100 percent.

While the worst case scenario is indicated when a given user can hardly execute most of the tasks assigned to, in most cases he can start a task but he may never be able to execute a task to its logical conclusion, then his learnability scale will be ranked as poor, this implies the user is not familiar with that particular software and the level of utilization is poor as probably the software may be new in the market or may have a complicated user interface.

In this tool the learnability of the user has a minimum value of 28 percent, this means when the user utilizes the social software ,the user is aware of different applications and those applications that have similarities in functionality, further the program, images and icons used can guide or prompt the user to its right usage however he may not recall the most commands used as they may be different as compared to other applications hence making the memorability to have a higher lower value and generally learnability will be have lower value.

In this case after adding all sub metrics and applying the learnability formula it will translate to maximum of 28 percent, the computations are demonstrated below:

Learnability = 4.149+ ((User interface features)7 x 0.013) + ((Program complexity) 7 x 0.264) + ((Memorability) 7 x 1.199) + 1.763 =28.4

The scale can be presented in Figure 1

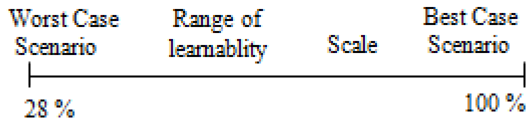

Figure 1: Scale of Learnability Assessment Model

The learnability assessment tool considers that the fact the best scenario when the user is able to login in to the system and do all tasks assigned. This means even if the given software or task assigned to is difficult the user is able to accomplish that task in the stipulated time frame and the learnability curve will at the highest level in a graph, further it does not consider how many times the user has utilized the software.

Worst case scenario is when the user is not able to accomplish any task assigned to within stipulated time to due to the low memorability of the software, due to complicated interface that makes the user difficult to remember or difficult language that make the language not easily remembered by the user. The model does not consider how long the user has utilized the given software.

For the purpose of interpretation of the results for mobile social software learnability the scale was transformed into Likert scale as presented in Table 1

Table 1: Interpreting Results for Mobile Social Software Learnability

Poor	Fair	Good	Very Good	Excellent
0-28	29-46	47-64	65-82	83-100

The learnability is presented in terms of percentages which indicates the number of tasks which a user is able to execute, this percentage has the range, where by the range indicates the total number of points which a given user has scored for instance 0-28 it means the user can hardly execute any task assigned to indicating has low learnability of the given software which is using to execute that task.

When the user executes a number of tasks assigned to then the learnability increases and it can fall in subsequent scales from fair to very good , but when the user executes all tasks within the given time limit then his learnability is said to be excellent.

3. Methodology

The paper adapted mixed research design. The primary data used in the paper was collected from a survey, targeting mobile social software users. Survey was used as it allows you to measure the significance of the mobile social software on the overall population, the target population was 6,000 and the sample size 361 of respondents was selected.

Experiment were also used to test the memorability of the mobile social software users by assigning the users specific tasks to perform the sample size of 30 respondents was selected. The study achieved 95.3% response rate of the target. This response rate was considered appropriate for analysis and reporting as supported by [16]. indicating that response rate of 90% and above is excellent. Descriptive statistics were computed, the results are tabulated in the next section.

4. Results

As can be seen in Table 2, User Interface Features, Program complexity and Memorability had positive regression weights of 0.013, 0.264 and 1.199 indicating respondents with higher scores on this scale were expected to have higher influence on Learnability, after controlling for the other variables in the model. The dependent variable for the model is Learnability, during this analysis significance was at 95% confidence level. Hence the three variables were used to design the tool. Learnability weights were computed and results are tabulated as shown in Table 2.

Table 2: Learnability Weights

Model	Unstandardized Coefficients		Standardized Coefficients	t	Sig.	Confidence Interval for B	
	B	Std. Error	Beta			Lower Bound	Upper Bound
(Constant)	4.149	0.650		6.39	0.000	2.871	5.427
Customization	-0.042	0.067	-0.036	-0.64	0.525	-0.174	0.089
Satisfaction	-0.026	0.064	-0.022	-0.40	0.691	-0.152	0.101
Software consistency	-0.008	0.064	-0.007	-0.12	0.903	-0.134	0.118
User interface features	0.013	0.026	0.011	0.50	0.620	-0.039	0.065
Language complexity	0.264	0.029	0.234	9.03	0.000	0.206	0.321
Memorability	1.199	0.035	0.769	34.49	0.000	1.130	1.267
a. Dependent Variable: LERNABILITY							

5. Conclusion

The findings for determining learnability factors affecting mobile social software applications, indicated that there exist a strong positive and statistically significant relationship between user interface features and learnability, The findings further indicate that there exist a positive and statistically significant relationship between social software program complexity and learnability, this implies that as the user uses the social software the program in the social software becomes easy and the users' guide becomes clear and easy to understand thus learnability increases in equal measure. The research also found that memorability has a strong positive and statistically significant relationship as per the results of the research, this implied that users who can remember how to perform an operation with ease have high memorability and this improves with time as the user utilizes the social software. The tool was designed.

References

[1] Calayag, M. B., Regala, D. M. M., Rivera, C. J. R., & Tancio, D. M. N. (2019). A comparative study of different automated teller machine and its effectiveness based on customer's satisfaction. *Journal of Industrial Engineering and Management Science*, *2019*(1), 49-102.

[2] ARIF, M. D. (2018). *Exploring Language Awareness of TEFL Graduate Students: A Case of Explicit Knowledge in Advanced Grammar Exam* (Doctoral dissertation, Universitas Sebelas Maret).

[3] Soylu, A., Kharlamov, E., Zheleznyakov, D., Jimenez-Ruiz, E., Giese, M., Skjæveland, M. G., ... & Horrocks, I. (2018). OptiqueVQS: a visual query system over ontologies for industry. *Semantic Web*, *9*(5), 627-660.

[4] Li, Y., & Zhu, L. (2019). Optimization of user experience in interaction design through a Taguchi-based hybrid approach. *Human Factors and Ergonomics in Manufacturing & Service Industries*, *29*(2), 126-140.

[5] Faizan, N., Gottlieb, M., Löffler, A., Utesch, M. C., & Krcmar, H. (2019, April). State-of-the-Art to Measure the TPACK Level of Trainees in Higher Education to Increase the Learnability of the Train-The-Trainer (TTT) Sessions. In *2019 IEEE Global Engineering Education Conference (EDUCON)* (pp. 384-391). IEEE.

[6] Lau, T. M. (2019). An Inclusive Design Guideline For Designing Elderly Friendly Smart Home Control Device.

[7] Rowsell, D. (2018). *Exploring Gay Men's Use of People-Nearby Applications* (Doctoral dissertation, Université d'Ottawa/University of Ottawa).

[8] Clarke, A. (2018). *Designing computer-based learning materials*. Routledge.

[9] Stenn, T. L. (2019). Building Resilient and Meaningful Enterprises with the Sustainability Lens. *Journal of Strategic Innovation and Sustainability*, *14*(3).

[10] Inzlicht, M., Shenhav, A., & Olivola, C. Y. (2018). The effort paradox: Effort is both costly and valued. *Trends in cognitive sciences*, *22*(4), 337-349.

[11] Ellis, K., Ritchie, D., Solar-Lezama, A., & Tenenbaum, J. (2018). Learning to infer graphics programs from hand-drawn images. In *Advances in neural information processing systems*(pp. 6059-6068).

[12] Masese Nelson, Samwuel Mbugua, Geoffrey Muchiri (2017). Research data for PhD thesis.

[13] Chatfield, C. (2018). *Introduction to multivariate analysis*. Routledge.

[14] Basili, V. R. (1992). Software modeling and measurement: the Goal/Question/Metric paradigm..

[15] von Wangenheim, C. G. (2018). *Thaisa Cardoso Lacerda*(Doctoral dissertation, Universidade Federal de Santa Catarina).

[16] Mugenda, O. M. (1999). Research methods: Quantitative and qualitative approaches. African Centre for Technology Studies.

Author Profile

Mr. Masese Nelson holds Phdstudent from Kibabii university , masters of computer applications degree from Periyar university , Bachelor of Science Information Technology from Sikkim Manipal university, currently His research interest are mobile applications IOT and cloud computing security.