An Adjustable Sample Size Estimation Model for Usability Assessment

[1]Haidar S. Jabbar, [1]T. V. Gopal and [2]Sattar J. Aboud
[1]Department of Computer Science and Engineering, College of Engineering,
Anna University, Chennai 600-025, INDIA
[2]Department of Computer Information Science, Faculty of IT,
The University of Gradate, Studies, P.O. Box 42 -11610, Amman, JORDAN

Abstract: Usability assessment plays a fundamental role in discovering usability problems and the determination of the level of usability for a given software product. One crucial aspect in every usability assessment is the estimation of the sample size desired for a software product. Once we start estimating the sample size needed for a usability assessment, a baffling question comes in mind: "how many users are enough for a given software product?". To the best of our knowledge, the majority of existing models estimates the sample size needed for a usability assessment, based on historical data. A better estimation should be based on both historical data, which provide an initial idea from previous software products and based on present data, which provide a practical idea for a given software product. Therefore, in this paper we have proposed an adjustable sample size estimation model for usability assessment, which enhances the estimation process by using two factors: Alpha factor (α), which estimates the problem discovery rate (λ_α) from historical data and the Beta factor (β), which estimates the problem discovery rate (λ_β) by using the complexity of the software product. λ_β is taken as vertical and domain wise for a given software product, to adjust the alpha factor appropriately to give the desired confidence level in the results. An illustrative case study has been provided at the end of the paper.

Key words: Usability Asssessment, Sample Size Estimation Models, Adjustable Sample Size Estimation Model, Alpha Factor (λ_α), Beta Factor (λ_β), Ilustrative Case Study.

INTRODUCTION

With the increase demand for software products and websites, the focus on the usability for these needs becomes essential [1,2]. Several studies in the field of usability engineering, have shown that in addition to functionality and other quality attributes, usability is a very important success factor with the huge delivery of diverse interfaces [3,4]. Moreover, usability is increasingly recognized as a vital quality factor for interactive software systems, since most researchers agree that unusable systems are probably the single largest reasons why encompassing interactive systems, computers plus people, fail in their actual use [5].

At the heart of usability engineering, usability assessment plays a fundamental role in discovering usability problems and benchmarking the level of usability for the software products [6]. One crucial related challenge, for any usability assessment study, is the estimation of the sample size (number of users

needed for the usability assessment) [7]. Therefore, when speaking about usability in relation to the sample size, a direct question comes in mind, "*how many users are enough for a given software product?*" Is it large sample size like 50, 100 150…etc, such as those estimated from typical research studies, or is it small sample size like 5, 8, 15...etc, such as those recommended by usability researchers [8].

Throughout the literature, sample size estimation for usability assessment has been done either as simply *guessing*, or using a *mathematical formula* proposed by the researchers. A variety of international research work has been done on this topic, especially by the researchers: Virzi [9], Nielsen and Launder [10] and Lewis [11]. However, the researchers presented a mathematical model of the problem discovery rates in usability assessment [12]. Using the problem discovery rate, research has shown that it was possible to estimate the sample size needed to uncover a given proportion of problems in an interface [9].

Corresponding Author: Haidar S. Jabbar, Department of Computer Science and Engineering, Anna University, Chennai 600025, INDIA

In this paper, sample size for typical research studies is not the same for usability assessment studies and then again sample size for usability assessment studies is not the same for all software products, conversely *"no size fit all"*. Therefore we have proposed an adjustable sample size estimation model for usability assessment, which is based on:

- Estimating the *historical* problem discovery rate (λ_α), which is already recommended by the researchers.
- Estimating the *adjustment (vertical / domain wise)* problem discovery rate (λ_β), by conducting a pre-assessment usability study, to gain a practical idea about the complexity for a given software product.

Integrating both points, gave us an improved and normalized estimation model toward the sample size for the usability assessment studies, as described below in formula (1):

$$SS_{EST\text{-}Adjustable} = \alpha + \beta \qquad (1)$$

where Alpha factor (**α**) is the initial historical estimation value based on λ_α and Beta factor (**β**) is the adjustment (vertical / domain wise*)* value based on λ_β. α is estimated from the historical problem discovery rate either taken as a constant (0.31) or from a set of individual unique problem discovery rates of similar software products (indicated in point 1 above) and β is estimated from the specific discovery rate for a given software product (indicated in point 2 above).

The proposed model is an outcome to improve the estimation of the sample size for usability assessment by balancing both historical and specific problem discovery rates for a given product to reach an accurate estimation of the sample size. This paper has been divided to 5 sections. In section 2, we have offered a brief overview of the general sample size estimation methods, the reasons for small usability assessment sample size and the factors that affect the selection of the usability assessment sample size. In section 3, we have discussed the needs of our approach and we have described our proposed estimation model for the usability assessment sample size. In section 4, we offered an illustrative case study for simulation purpose. Eventually, in section 5, we have concluded concerning our work and point toward the basis for future work.

MATERIALS AND METHODS

General Estimation of the Sample Size: A common goal of any research is to collect data representative of a population [13]. However, the estimation of sample size is not always a common and direct task for many organizations and researchers. It is a crucial step, since inappropriate, inadequate, or excessive sample sizes continue to influence the quality and accuracy of the research [14].

It is useful to discuss the difference of the sample size for typical research and for usability assessments. In a typical research, a representative sample of people are asked a small controlled set of questions about subjective thoughts and feelings, especially opinions regarding liking or disliking a certain product, interaction or political position. Such research aim for statistically-valid sample sizes where the confidence in the results can be calculated using standard formulas [15].

Usability assessment, on the other hand, aims to measure users' performance, not opinions. A representative user in a well-designed experimental environment is asked to perform specific tasks or operations. Usability assessments focus on the goal of task success, not user opinion. Quantitatively speaking, the sample size for typical research is always large; probably more than 50, i.e. a size of an institution or city and so on, while for usability assessment studies it is small probably 5 to 15 users. According to previous studies, projects and discussions by many researchers [16], approximately (*6-7*) users is the optimized sample size for usability studies, where the benefits and resources meet (focal point), above that or below will diminish, since cost increase or assessment quality decrease, therefore usability assessment should balance both resources and benefits by not conducting expensive and time consuming usability assessments, it should be simple and effective, as shown in figure-1 below, the *slope* of the curve define this (focal point):

There are many reasonable factors that diminish the sample size of usability assessment in contrast with typical research [17]:

- There is less variability among human subjects in task performance than in subjective opinions [1].
- A successful task is made up of successful sub-tasks. Problems, and thus solutions, relating to sub-tasks will have an impact beyond the specific scenario in which they are assessed. In a well designed usability assessment, the same ergonomic principle can often be assessed several times in a single interaction.

Trained analysts study usability results, looking for patterns of behavior which are symptomatic of

known usability problems. Thus the experience of many previous assessments and fielded systems is brought to bear on the current assessment, dramatically boosting the effective sample size for a given test.

- In general, user interface devices tend to work well across all subjects, or they tend to fail across all or most subjects. When this is coupled with
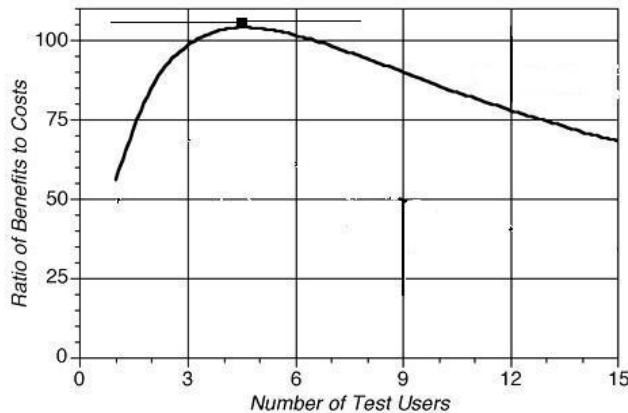

Fig.1: Ratio of Benefits/Costs to the Sample Size

observations 1) and 2) and 3) above, this leads to reliable results with a relatively small set of subjects.

- Large sample sizes do increase statistical confidence, but this does not help with either identifying the root cause of the problem or the effect of changing the device across the population.
- Usability assessments are about ergonomic design. They do not attempt to quantify or predict population statistics. Larger sample sizes are therefore simply more expensive, and rarely if ever, more effective [18].
- Resource limitations, where here we should remember that 1 hour of video for usability assessment, needs about 8 hours of analysis [19].

Generally, there are 2 main sample size estimation methods for research studies, which are mainly based on the collected data types: one is for (continuous and ordinal data) and the other for discrete *(categorical data)* [20]. Both have their own formulas for the estimation of the sample size. Since they are used in our proposed model, a short overview has been provided below in the following sections, to gain a rough idea about these sample sizes.

The Sample Size for Discrete Data: Nearly most of typical research studies use this type of sample size, where the quantity of the users plays a critical role in the determination research results. If we look at this type in detail, we notice that the sample size for this type is usually high. For this type, a representative sample of people are asked a small controlled set of questions about subjective thoughts and feelings, especially opinions regarding liking or disliking a certain product, interaction or political position. The questions are binary type i.e.: (yes, no), (like, dislike) and (0, 1).

This type of sample size can be estimated from the formula (2) below:

$$SS_{EST\text{-}DD} = \frac{(t)^2 * (p) * (q)}{(d)^2} \qquad (2)$$

t is alpha value at confidence level (95%), **p** and **q** are the variance, and **d** is the confidence interval. This formula is taken from Chocran [21]; on the other hand there is other formulas proposed by different researchers for this type of sample size; however there is slight variation among them. As discussed early this type is not appropriate for our study (usability assessment), since we are looking for the *quality not the quantity* of the users interacting with a software product, therefore if we look at the formula we can realize that (p) and (q) values are computed from the variance of the answered (questions), which produced discrete (binary) data, which do not comply with the data type generated from main usability assessment studies. For usability assessment, we are looking for the significance of the standard deviation among users performance that precisely fit with our usability assessment data.

The Sample Size for Continuous Data: This type of sample size estimation is different from the previous, where the *quality* of the users plays a critical role in the determination of the research results. If we look in detail at this type, we notice that the sample size for this type usually is low. For this type, a representative sample of users is given questions which measure their performance and quality interacting with the system. The questions are continuous or ordinal type i.e.: scale (1, 2, 3, 4, 5) and ratio (0-100%).

This type of sample size can be estimated from the formula (3) below:

$$SS_{EST\text{-}COD} = \frac{(t)^2 * (sd)^2}{(d)^2} \qquad (3)$$

t is alpha value at confidence level (95%), **sd** is the standard deviation of the population, and **d** is the confidence interval. This formula has also been taken from Chocran [21]; on the other hand there is other formulas proposed by different researchers for this type of sample size; however there is slight variation among them. This type of sample size has been employed in the estimation of β factor, since if we look at the formula we can realize that (sd) value is computed from the standard deviation of a continuous data type.

Therefore, we conclude that continues sample size, is more appropriate for usability assessment studies, since we are using the standard deviation of the users instead of the variance in formula (2), which is more close to the complexity of users' performance. The data generated with this method comply with the data type generated from most usability assessment studies. However, when speaking about usability, there are many factors that effect the variation of the estimation for the sample size, as discussed in [12]:

- Properties of the system and interface, including the size of the software product [22].
- Stage in the usability lifecycle the product is assessed in, whether early in the design phase or after several iterations of test and re-design.
- Type and quality of the methodology used to conduct the assessment.
- Specific tasks selected.
- Match between the assessment and the context of real world usage.
- Representativeness of the assessment users [23].
- Skill of the evaluator.

Based on the above discussion, we have realized that continues sample type is more appropriate from discrete sample type for the usability assessment, since data comply with the data generated from usability studies and there is enough rooming to represent the generated usability assessment data. In the following section, we have offered a detail description of our proposed model.

An Adjustable Sample Size Estimation Model For Usability Assessment: In this section, we have proposed our adjustable model for the estimation of usability sample size, which is based on α and β factors, as discussed early, in section 1. This model is proposed to balance between an initial estimation of the historical problem discovery rate and on a specific estimation of the problem discovery rate for a given software product. In summary, we are intending to find the sample size difference between the above 2 factors, we

represent this mathematically as the absolute difference of the 2 factors as:

$$\text{SS}_{\text{Difference}} = \frac{\Delta\alpha}{\Delta\beta} = \left| \lambda_\alpha - \lambda_\beta \right| \qquad (4)$$

To find this difference, we need compute 2 problem discovery rates: λ_α and λ_β, based on the factors α and β respectively. The alpha factor is taken from historical data either constant (0.31) or from a set of users and their individual problem discovery rates in similar software products. The beta factor may be taken "*vertical and domain wise*" and it adjusts the alpha appropriately to give the desired confidence level in the results. Thus, in the following sections, we have offered a detailed description of the proposed model, including the estimation of both sample size values and a way to adjust them, based on their problem discovery rates.

Initial Estimation of the Sample Size (λ_α) from Alpha (α) Factor: α is already explained in numerous research works mainly by Nielsen, Virizi and Turner and has been discussed early. However, this value is estimated from historical data λ_α (problem discovery rates). λ_α is either taken (0.31) as suggested by Nielsen [8] or could be estimated from previous historical data to replace the above value of λ_α. If we go for the historical data, then λ_α is estimated by creating a matrix of users' numbers and their related individual problem discovery rates.

From this matrix, we are able to estimate the problem discovery rate λ_α dynamically, rather than going for the constant (0.31), which is static for all software products. The main steps to estimate λ_α value from a set of users is given below and it is already described in [12].

After obtaining a set of problem discovery rates for each user, then we present this information as a matrix and then obtain an average rate of all users. Unless the sample size is small, we need to use some method to adjust this small value properly by using the good-turing discounting formula (5) and the normalization formula (6) slightly underestimated problem discovery rates, which is already explained in [12].

$$\lambda_{\text{GT-Adj}} = \frac{\lambda_{EST}}{\left(1 + \frac{O}{N}\right)} \qquad (5)$$

where λ_{EST} is the initial estimate computed from the raw data of a usability assessment study, O was the

number of usability problems detected by only one user, and N was the number of assessed users.

$$\lambda_{\text{NORM-Adj}} = \left(\lambda_{EST} - \frac{1}{n} \right) * \left(1 - \frac{1}{n} \right) \qquad (6)$$

where λ_{EST} is the initial estimate computed from the raw data of a usability assessment study and n was that total number of unique usability problems detected by all users. Applying each formula (5) and (6) to the initial estimate of λ_α, then averaging the results, produces a highly accurate estimate of the problem discovery rate. Eventually, formula (7) shows the formula for an adjusting λ_α based on averaging good-turing and normalization adjustments.

$$\lambda_\alpha = \frac{1}{2} \left(\lambda_{GT-Adi} + \lambda_{NORM-Adj} \right) \qquad (7)$$

where λ_α is the average of the problem discovery rate for historical data. Additionally, we can compute the number of users needed, based on λ_α value only by using formula (8):

Proportion of Unique Problems = $1 - (1 - \lambda)^n$ (8)

where λ in this section means λ_α the problem discovery rate from historical data. In the following section, we will see how to estimate the λ_β problem discovery rate for a given software product.

Adjustable Estimation of the Sample Size (λ_β) from Beta (β) Factor: β value is an adjustment factor, used to provide us crucial information related the sample size for a specific software product. λ_β is estimated by conducting a pre-assessment study of the highest task time of the software product. We have selected the (task time) to represent the software product complexity, because time always measures the whole complexity among other usability metrics [24]. The idea behind this short usability assessment is that we want to gain a practical idea of the product complexity that aids us in predicting the sample size for this particular software product. Usually we select (2-3) users to conduct this pre-assessment (which represent approximately 25% of the total sample size) with users having different experience, to reduce the probability of error in the confidence interval (d).

Now for computing β factor, we first start to conduct a pre-assessment usability study for the highest task time for 2-3 users. We need to keep in mind that those 2-3

users should differ in their experience, at least 1 novice and 1 experienced user, to estimate the maximum variance of time among users. Now, counting their task time's t1, t2…tn, next we need to compute the mean range of their task times, as shown in formula (9):

$$\text{Mean Range} = \frac{\sum_{i=1}^{n} |t_i - Range_t|}{n} \qquad (9)$$

where t_i is the instances of users times for a given task, $Range_t$ is the absolute difference between the maximum task and the minimum task time and n is the number of users.

At the present, this mean range time or a less time is the confidence interval value for that particular task. Within this value we start to match between the sample size and the confidence interval to reach an approximate number of users for a usability assessment, using the formula (3) of the continuous data method, reviewing above for this step we can ask the question in a different manner: rather than asking "how many users for a given product?" we ask "how many users for a given product with a desired confidence?". However, we need to keep in mind that here in this assessment, we do not have the complete sample of users, this means that we can not compute the standard deviation (sd) based on the sample, therefore we need to compute it from a small sample size, to solve this we will use the T-Distribution [25].

This distribution uses a value from T-table, which approximate the standard deviation value for this small sample size. Now altering places of the variables in formula (3), the confidence interval is computed as follows in formula (10):

$$d = \frac{f * sd}{\sqrt{SS_{EST-COD}}} \qquad (10)$$

f is t-value, **sd** is the standard deviation of the small sample size, **$SS_{EST\text{-}COD}$** is the number of users for the small sample and **d** is the confidence interval which is already known from the mean range of task time or less.

Usually, it does not take more than 5 simple computations to find an approximate sample size for a given software product. Unless we are obtaining a problem discovery rate λ_α for α, then we also need to obtain problem discovery rate for λ_β from β and then average them to find an adjusted λ value, which is based on the historical data as well as for a given software product. Now for computing λ_β (at 90% likelihood), we use the formula (11):

$$\lambda_\beta = 1 - \sqrt[n]{0.10} \qquad (11)$$

where n is the number of users and λ_β is the problem discovery rate for β factor.

At the present, we have 2 problem discovery rates, which are λ_α and λ_β, averaging them will give us an adjusted problem discovery rate (λ), given below in formula (12):

$$\lambda = \frac{1}{2}\left(\lambda_\alpha + \lambda_\beta\right) \qquad (12)$$

where λ_α is the problem discovery rate for α, λ_β is the problem discovery rate for β and λ is the average problem discovery rate. Eventually, the final step is to estimate the sample size for the usability assessment based on both historical data and for a given product (at 90% likelihood), given below in the formula (13):

$$SS_{EST-Adjustable} = \frac{-1}{\log(1 - \lambda)} \qquad (13)$$

where λ is the average problem discovery rate and $SS_{EST-Adjustable}$ is the final estimated sample size number for the usability assessment study for a given software product.

RESULTS AND DISCUSSIONS

An Illustrative Case Study: In this section, we have presented an illustrative case study for simulating the computations of our proposed model. As discussed early in section 3, λ_α is either taken as (0.31) as recommended by Nielsen or we can estimate it from previous study from a matrix of users and their individual problem discovery rates, as shown in (the presented case study values) table-1 below.

The average problem discovery rate for all users is computed below:

$$\lambda_{\alpha(1-6)} = \frac{.5 + .67 + .67 + .5 + .2 + .5}{6} = 0.51$$

Table 1: Usability Assessment Data for 6 Subjects and 6 Users

	USERS DATA						
E No.	U1	U2	U3	U4	U5	U6	
E1	1	1	1	0	0	1	
E2	0	0	1	1	1	0	
E3	0	1	1	0	0	0	
E3	0	0	0	1	0	1	
E4	1	0	0	0	0	0	
E5	1	1	0	1	0	0	
E6	0	1	1	0	0	1	
λ_α	0.5	0.67	0.67	0.5	0.2	0.5	.51

The average problem discovery rate ($\lambda_{\alpha(1-6)}$) for this previous software product is: (0.51).

Now, for improving the problem discovery rate λ_α, we need to use some method to adjust this small value properly by using the good-turing discounting formula (5) and the normalization formula (6) slightly underestimated problem discovery rates, as follows:

$$\lambda_{GT-Adj} = \frac{0.51}{\left(1 + \frac{1}{6}\right)} = .44$$

$$\lambda_{NORM-Adj} = \left(.51 - \frac{1}{6}\right) * \left(1 - \frac{1}{6}\right) = .28$$

The adjusted problem discovery rate is obtained by averaging the two estimates using formula (7) gives:

$$\lambda_{Adj} = \frac{1}{2}\left(.44 + .28\right) = .36$$

Now this λ_α value (0.36) will replace the standard value (0.31). With this adjusted value of λ_{Adj} and the known sample size, it is possible to estimate the sample size adequacy of this previous study (at 90% likelihood) using formula (8):

Proportion of Unique Problems (0.90) $= 1 - (1 - 0.36)^6$

We can observe that for the average likelihood (90%), on (0.36) average adjusted problem rate for λ_α approximate 6 users is enough for this previous software product.

Next we will estimate λ_β value. As discussed early in section 3.2, this value is estimated from a given specific software product. To start this, we need to conduct a pre-assessment study with 2-3 users. Let us assume that 3 users have conducted this pre-assessment study, 1 novice and 1 experienced and 1 moderate. They completed their task time on t1 (192), t2 (344) and t3 (143) respectively. For computing the task times mean range we will use formula (9), as shown below:

Mean Range $= \dfrac{|143 - 201| + |192 - 201| + |344 - 201|}{3} = 70\ Sec$

Based on the above results the confidence interval (range of task time) for this task will be 70 Sec, however we will selected 60 for better confidence interval. Since we do not know the standard deviation of the complete sample size, we need to estimate it from the small sample size; therefore instead of using a

z vale we will use a t value from a T-Distribution. We will make several attempts to reach an approximate match of the confidence interval:

Attempt 1: sd is 105, t value is 2.353 and n is 3, using formula (10), the confidence interval is:

$$d = \frac{2.353 * 105}{\sqrt{3}} = 142$$

Unless 142 is not equal to 60, we have to make increase the number of users for this task and make another attempt. **Attempt 2:** sd is 105, t value is 2.132 and n is 4, using formula (10), the confidence interval is:

$$d = \frac{2.132 * 105}{\sqrt{4}} = 112$$

Unless 112 is not equal to 60, we have to make increase the number of users for this task and make another attempt. **Attempt 3:** sd is 105, t value is 2.015 and n is 5, using formula (10), the confidence interval is:

$$d = \frac{2.015 * 105}{\sqrt{5}} = 95$$

Unless 95 is not equal to 60, we have to make increase the number of users for this task and make another attempt. **Attempt 4:** sd is 105, t value is 1.943 and n is 6, using formula (10), the confidence interval is:

$$d = \frac{1.943 * 105}{\sqrt{6}} = 83$$

Unless 83 is not equal to 60, we have to make increase the number of users for this task and make another attempt. **Attempt 5:** sd is 105, t value is 1.895 and n is 7, using formula (10), the confidence interval is:

$$d = \frac{1.895 * 105}{\sqrt{7}} = 75$$

Unless 75 is not equal to 60, we have to make increase the number of users for this task and make another attempt. **Attempt 6:** sd is 105, t value is 1.860 and n is 8, using formula (10), the confidence interval is:

$$d = \frac{1.860 * 105}{\sqrt{8}} = 69$$

Unless 69 is not equal to 60, we have to make increase the number of users for this task and make another attempt. **Attempt 7:** sd is 105, t value is 1.83 and n is 9, using formula (10), the confidence interval is:

$$d = \frac{1.83 * 105}{\sqrt{9}} = 64$$

Unless 64 is not equal to 60, we have to make increase the number of users for this task and make another

attempt. **Attempt 8:** sd is 105, t value is 1.83 and n is 9, using formula (10), the confidence interval is:

$$d = \frac{1.812 * 105}{\sqrt{10}} = 60.04$$

Unless 60.04 is approximately equal to 60 (60.04≈60), then the recommended number of users for this given product is 10 users. Now we need to estimate λ_β value. Unless we know all the variables, then λ_β can be estimated from formula (11):

$$\lambda_\beta = 1 - \sqrt[10]{0.10} = 0.20$$

where n is 10 users. Now we have 2 problem discovery rates, which are λ_α and λ_β, averaging them will give me an adjusted problem discovery rate (λ), given below using formula (12):

$$\lambda = \frac{1}{2}(0.36 + 0.21) = 0.28$$

Eventually, the final step is to estimate the sample size for the usability assessment based on both historical data and for a given complexity of a software product (at 90% likelihood), given below in the formula (13):

$$SS_{EST-Adjustable} = \frac{-1}{\log(1 - 0.28)} = 7.1$$

SS$_{EST–Adjustable}$ is the final normalized sample size number for the usability assessment study for a given software product. For this software product, we can say the best estimation for the sample size is about **(8)** users.

CONCLUSION

As on date, there is no unique model for the sample size estimation of usability assessment. The majority of existing models estimate the sample size needed for a usability assessment, based on historical data. In this paper, we have extended this notion by proposing a specific adjustable factor (λ_β) that adjusts the initial sample size estimation appropriately to give the desired confidence level for the results.

The proposed model is based on both historical data of the problem discovery rate (λ_α) and on specific software product problem discovery rate (λ_β). λ_α is either taken (0.31) or estimated from a set of users and their individual problem discovery rates. λ_β is estimated from conducting a pre-assessment usability study to gain an practical idea that aid us to predict the complexity of the software product. Eventually, λ is an adjusted value based on the average of both λ_α and λ_β.

Both theoretical and the illustrative case study discussed above, indicate that the proposed adjustable model normalize and compensate the estimated problem discovery rate (λ). However, considerable work still remains to be carried out and to be verified for answering the questions about when usability assessment is dependable, accurate, and effective.

REFERENCES

1. Rubin J., 1994. Handbook of Usability Testing: How to Plan, Design and Conduct Effective Tests. John Wiley and Sons.
2. Haidar S. Jabbar and T. V. Gopal, 2006. User Centered Design for Adaptive E-Learning Systems. Asian Journal of Information Technology, 5, pp. 429-436.
3. Nielsen J., 1993. Usability Engineering. Boston Academic Press.
4. Patrick W. Jordan, 1998. An Introduction to Usability. Taylor and Francis.
5. Ahmed Seffah, Mohammad Donyaee, Rex B. Kline and Harkirat K. Padda, 2006. Usability Measurement and Metrics: A Consolidated Model. Software Quality Journal, 14, pp. 159-178.
6. Haidar S. Jabbar and T. V. Gopal, 2006. An Integrated Metrics Based Approach for Usability Engineering. International Review of Computers and Softwares, 1, pp. 114-123.
7. Carol M. Barnum, 2003. What's in a Number?. STC Usability SIG Newsletter, 3 (9).
8. Nielsen J., Why You Only Need to Test with 5 Users, retrieved from: http://www.useit.com/alertbox/20000319.html.
9. Virzi R.A., 1992. Refining the Test Phase of Usability Evaluation: How Many Subjects is Enough?. Human Factors, 34, pp. 457-468.
10. Nielsen J. and Landauer T.K., 1993. A mathematical Model of the Finding of Usability Problems, In Proceedings of ACM INTERCHI'93 Conference, Amsterdam, Netherlands: ACM Press, pp. 206-213.
11. Lewis J.R., "Sample Sizes for Usability Studies: Additional considerations", Human Factors, Vol. No. 36, pp. 368-378, 1994.
12. Turner C. W. Turner, Lewis J. R. and Nielsen J., 2006. Determining Usability Test Sample Size. International Encyclopedia of Ergonomics and Human Factors, 3, pp. 3084-3088.
13. Bartlett J. E., Kotrlik J. W. and Higgins C., 2001. Organizational Research: Determining Appropriate Sample Size for Survey Research. Information Technology, Learning, and Performance Journal, 19, pp. 43-50.
14. Ary D., Jacobs L. C. and Razavieh, 1996. A Introduction to Research in Education. Fort Worth, TX: Harcourt Brace College Publishers.
15. McGee M., 2004. Master Usability Scaling: Magnitude Estimation and Master Scaling Applied to Usability Measurement, In Proc. CHI, Washington, D.C., ACM Press, pp. 335-3420.
16. Nielsen J., 1989. Usability Engineering at a Discount in G. Salvendy and M.J. Smith (Eds.), Using Human-Computer Interfaces and Knowledge-Based Systems. Amsterdam: Elsevier, pp. 394-401.
17. Enterprise Integration Group, Usability Testing and Sample Sizes: Best Practice Recommendations, retrieved from: hhttp://www.eiginc.com/documents/Services/UsabilityTesting.pdf.
18. Caplan Stanley, 2004. Ergonomic Guidelines for Designing Machines. Machine Design, 6, page 53.
19. Deborah J. Mayhew, 1999. The Usability Engineering Life Cycle. Morgan Kauffman Publisher.
20. Stephan H. Kan, 1995. Metrics and Models in Software Quality Engineering. Addison Wesley.
21. Cochran W. G., 1977. Sampling Techniques. New York: John Wiley and Sons.
22. Richard H. Spencer, 1985. Computer Usability Testing and Evaluation. Prentice Hall.
23. Andrew Dillon and Min Song, 1997. An Empirical Comparison of the Usability for Novice and Expert Searchers of a Textual and a Graphic Interface to an Art-Resource Database. Journal of Digital Information, 1, pp. 09-26.
24. Kasper Hornbaek, 2006. Current Practice in Measuring Usability: Challenges to Usability Studies and Research. International Journal of Human-Computer Studies, 64, pp. 79–102.
25. M. Abramowitz and I. A. Stegun, 1972. Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. New York: Dove.