Proceedings of the Human Factors and Ergonomics Society 42nd Annual Meeting
(Chicago, October 5-9, 1998), pp. 1336-1340. HFES, Santa Monica, CA.
Preprint version

THE EVALUATOR EFFECT IN USABILITY STUDIES: PROBLEM DETECTION AND SEVERITY JUDGMENTS

Niels Ebbe Jacobsen
Morten Hertzum
University of Copenhagen
DK-2300 Copenhagen, Denmark

Bonnie E. John
Carnegie Mellon University
Pittsburgh, PA 15213-3891

Usability studies are commonly used in industry and applied in research as a yardstick for other usability evaluation methods. Though usability studies have been studied extensively, one potential threat to their reliability has been left virtually untouched: the evaluator effect. In this study, four evaluators individually analyzed four videotaped usability test sessions. Only 20% of the 93 detected problems were detected by all evaluators, and 46% were detected by only a single evaluator. From the total set of 93 problems the evaluators individually selected the ten problems they considered most severe. None of the selected severe problems appeared on all four evaluators' top-10 lists, and 4 of the 11 problems that were considered severe by more than one evaluator were only detected by one or two evaluators. Thus, both detection of usability problems and selection of the most severe problems are subject to considerable individual variability.

INTRODUCTION

A usability study – also known as a think-aloud study – is probably the single-most important usability evaluation method (UEM) in practice (Nielsen, 1993), and it is undoubtedly the most investigated UEM. Many dimensions of usability studies have been investigated including the sufficient number of users (e.g. Lewis, 1994; Virzi, 1992), individual versus cooperating users (Hackman & Biers, 1992), the level of experimenter intervention (Held & Biers, 1992), task settings (Held & Biers, 1992; Karat et al., 1992), retrospective versus concurrent think-aloud (Ohnemus & Biers, 1993), and the impact of usability tests in real-life settings (e.g. Jørgensen, 1989). Moreover, usability testing has been compared to and used as a yardstick for other UEMs (see e.g. Bailey et al., 1992; Cuomo & Bowen, 1994; Henderson et al., 1995; John & Marks, 1997; John & Mashyna, 1997; Karat, 1994).

However, the effect of the *evaluator* on the process has been left virtually untouched in usability tests, although it has been studied in other UEMs such as Heuristic Evaluation (Nielsen & Molich, 1990). For example, in his chapter entitled "Usability Testing", Nielsen (1993) discussed the effect of variability in *users* as the *only* threat to the reliability of usability tests. Furthermore, Virzi et al. (1993) state that the "...think-aloud method incorporates the users' perspectives directly, without being filtered by an intermediary..." (p. 312) On the other hand, Holleran (1991) observed that there can be substantial disagreement among evaluators because the collected data are primarily subjective in nature, but he supplied no data to confirm this assertion. This paper extends the study of Jacobsen et al. (1998) in that it addresses how the detection and severity rating of usability problems depend on the evaluators who observe and analyze the usability test sessions. We focus on a quite controlled variant of usability tests where identical test sessions are conducted in a usability lab, under the management of an experimenter who only interferes in the user's work if strictly necessary.

METHOD

The usability test sessions

Four experienced Macintosh users spent about an hour thinking aloud as they individually worked through a set of tasks in a multi-media authoring system hereafter called the *Builder* (Pane & Miller, 1993). None of the users had previous experience with the Builder, and they did not receive any instructions in the use of the system. The Builder resembles an advanced word processor in that the user can create documents consisting of plain text, still graphics, movies, and animations.

The users were asked to create a new document based on a printed target document consisting of pages containing text, figures, and animations. The users had to add and edit some glossary items, add entries to a table of contents, delete a page, switch two pages, and save their document in two different versions. The same experimenter ran all four sessions. He did not interrupt the users unless they forgot to think aloud, explicitly gave up solving a task, or got stuck for more than three minutes. Prior to each session the experimenter introduced the study, taught the user to think out loud, and handed out the first task to the user after having read it out loud. Whenever the user finished a task, the experimenter handed out the next task. The sessions were videotaped for later analysis. In the following all users will be addressed as “he” though both males and females participated.

Evaluators

Four HCI research evaluators, all familiar with the theory and practice of usability testing, analyzed the four videotapes. Table 1 shows the evaluators’ experience with the Builder and their evaluation experience in terms of the total number of users previously analyzed. The authors of this paper were themselves evaluators in the study (three of the four evaluators). The third author designed the usability study, but the experimenter who ran the test sessions was not aware of our evaluator-effect research. The first two authors, who had no input into the design of the usability study, conducted the compilation and analyses of the evaluator’s responses. In the following all evaluators will be addressed as “she” though both males and females participated.

Eval-uator	Occupation	Number of users previously analyzed	Initial experience with the Builder	Average analysis time per tape
E1	Associate professor	52 users	10 hours	3.8 hours*
E2	Doctoral student	4 users	5 hours	2.7 hours
E3	Assistant professor	6 users	2 hours	2.9 hours
E4	Usability lab manage	66 users	12 hours	4.5 hours

Table 1. The HCI research evaluators’ previous usability test experience, their experience with the Builder and the average time spent analyzing each tape (each tape lasted approximately 1 hour). *The analysis time shown for E1 is the time spent analyzing the last tape, as she did not keep track of the time she spent on the first three tapes.

Procedure

Evaluators E1 and E2 knew the Builder well before this study was conducted; evaluators E3 and E4 familiarized themselves with it prior to their analysis. All evaluators had access to a written specification of the Builder (35 pages) and the running system throughout their analysis. The evaluators were asked to detect and describe all problems in the interface based on analyzing the four tapes in a preset order. No time constraints were enforced (see Table 1 for the time spent analyzing a tape). The evaluators were requested to report three properties for each detected problem: (a) a free-form problem description, (b) evidence consisting of the user’s action sequence and/or verbal utterances, and (c) one of nine predefined criteria for identifying a problem.

The evaluators used the following set of problem detection criteria: (1) the user articulates a goal and cannot succeed in attaining it within three minutes, (2) the user explicitly gives up, (3) the user articulates a goal and has to try three or more actions to find a solution, (4) the user creates an item in his new document different from the corresponding item in the target document, (5) the user expresses surprise, (6) the user expresses some negative affect or says something is a problem, (7) the user makes a design suggestion, (8) the system crashes, and (9) the evaluator generalizes a group of previously detected problems into a new problem.

Using the four evaluators’ individual problem reports (276 raw problem reports), the first two authors (NJ and MH) created a master list of unique problem tokens (UPTs) using the following procedure. First, each author split apart any of the original raw problem reports he thought contained more than one problem. NJ split 16 original reports, producing an additional 23 problems; MH split 17 original reports, producing an additional 18 problems. Eight of these new problem reports were the same, so both authors had a list of 284 problems in common (with MH having an additional 10 and NJ having an additional 15). Each author then examined their lists and eliminated duplicates. Of the 284 problems on both lists, the authors agreed on 245 (86%) as to whether they were unique or duplicated. The authors discussed the disagreements and the problems they did not share, reached consensus, and formed a master list of 93 UPTs.

To study the evaluators’ judgment of problem severity the evaluators received a version of the master list containing (1) a short description of each UPT, (2) the number of users experiencing the UPT, (3) the number of evaluators detecting it, (4) the problem detection criteria it was attributed to, and (5) the interface feature it involved. Each evaluator was presented with a scenario in which a project manager had constrained the evaluators to point out the ten most severe UPTs, as a tight deadline forced the developer team to fix only those few UPTs in the next release of the Builder. In the scenario the evaluators were told that their selection of UPTs should be based on the information on the master list and on other factors, such as considerations concerning experienced versus novice users, and the Builder’s use in real life settings. The UPTs on the top-10 lists were not prioritized, but each UPT was annotated with the evaluator’s reasons for including that particular UPT.

RESULTS

The percentages of the total of 93 UPTs reported by E1, E2, E3, and E4 were 63%, 39%, 52%, and 54% respectively. Thus, a single evaluator detected on average 52% of all known UPTs in the Builder interface.

The effect of adding more evaluators to a usability test resembles the effect of adding more users; both additions increase the overall number of UPTs found. Figure 1 depicts the number of the 93 UPTs detected as a function of the number of both evaluators and users. The average increase in UPTs found was 46% going from one to two evaluators, 23% going from two to three evaluators, and 17% going from three

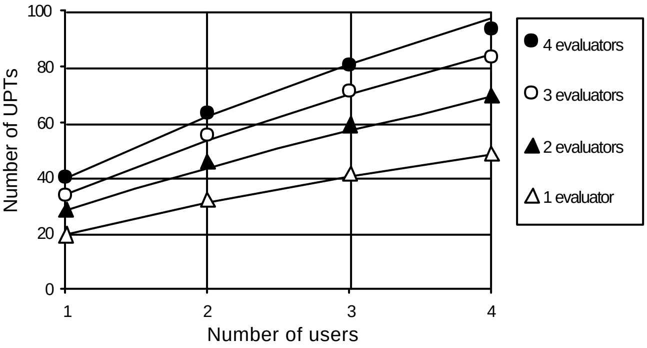

Figure 1. The number of detected UPTs depends on the number of users and the number of evaluators. The data points are the observed numbers, averaged over all combinations of users or evaluators. The curves plot Equation 1 for 1, 2, 3, and 4 evaluators.

to four evaluators when all four user-videotapes were included in the calculation (the four points on the rightmost vertical). Calculating the effect of running more users, we found an increase of 55% going from one to two users, 26% going from two to three users, and 23% going from three to four users when all evaluators were included in the calculation (the topmost curve). The declining number of new UPTs detected as more users are added confirms the results from similar studies (Lewis, 1994; Nielsen & Landauer, 1993; Virzi, 1992).

Our data can be described with Equation 1. The fit between the equation (the curves in Figure 1) and our data (the data points in Figure 1) is highly significant (squared correlation coefficient (R^2) = 0.997; standard error of estimate = 2.6%; $p<0.001$). The equation describes the relationship between the number of detected UPTs, number of users, and number of evaluators for our study. Other studies may result in different values for the constant and the exponents.

$$\text{No. of UPTs} = 19.35 * (\text{no. of evaluators})^{0.505} * (\text{no. of users})^{0.661} \quad (1)$$

The evaluator effect for all UPTs is substantial; as much as 46% of the UPTs were found by only a single evaluator, while 20% were found by all four evaluators (see Figure 2). Problem criteria 9 (a problem identified as a generalization of previously detected problems) might be more likely to differ across evaluators, since the generalization process is quite subjective. However, only 5% of all problem reports were attributed to criteria 9. Hence the evaluator effect cannot be caused by this criteria alone.

To investigate whether the level of agreement among the evaluators differs when detecting more severe problems, we used three methods to extract severe problems. First, we extracted the 37 UPTs attributed, by any evaluator, to one or more of the three problem criteria we thought more severe than the rest: (1) the user articulates a goal and cannot succeed in attaining it within three minutes, (2) the user explicitly gives up, and (8) the system crashes. Second, we looked at the 25 UPTs that appeared on at least one evaluator's top-10 list. Third, we extracted the 11 UPTs that were included on more than one top-10 list. Table 2 shows that the evaluator effect in detecting problems was progressively less extreme for the sets of more severe problem, but even for the smallest set of severe problems it was still substantial.

UPTs	No. of UPTs	Detected by			
		only 1	any 2	any 3	all 4
		evaluators			
All UPTs	93	46%	20%	13%	20%
Violating criteria 1, 2, or 8	37	22%	19%	19%	41%
Any UPTs on top-10 lists	25	20%	20%	80%	52%
More than one top-10 list	11	9%	27%	0%	64%

Table 2. Percentages of the UPTs detected by only 1, any 2, any 3, and all 4 evaluators.

Detection is not the only measure of interest, however. Severity judgment also differed substantially between the four evaluators. Looking at their top-10 lists, we found large differences; 56% of the 25 UPTs that appeared on the four top-10 lists were selected by only a single evaluator, 28% were selected by two evaluators, and 16% were selected by three evaluators. Not a single UPT appeared on all four top-10 lists!

DISCUSSION

Previous studies have shown that no single user will come across all problems in an interface. Our study refined this finding by suggesting that no single evaluator will detect all problems in a usability test. The number of problems revealed in a usability test is dependent on both the number of users and the number of evaluators.

Using Equation 1, we can estimate how many new UPTs a fifth user might find. In our study, five users and one evaluator could be traded for three users and two evaluators without decreasing the number of detected problems. Moreover, the two evaluators analyzing three users will, on average, detect the same number of problems from the union of the top-10 lists as the single evaluator analyzing five users.

Using different approaches to identify the severe problems in the Builder interface we found that more severe problems showed a tendency toward being detected by more evaluators. This tendency is in keeping with Virzi's (1992) results, but given the lack of statistical power in our study it does not contradict Lewis's (1994) result of no significant correlation between problem detection and problem severity.

Problem severity can be judged by the evaluators who initially detected problems in the interface or by a different group of people not affected by the process of detecting problems prior to their severity judgment. The evaluators who initially detected problems in the interface will be fully able to understand the problem descriptions and the interface as they have worked closely with the interface and the videotapes prior to their severity judgment. However, such evaluators may be biased toward the problems they originally detected themselves. Jeffries (1994) found that problem reports are often unclear and ambiguous. Hence, relying on severity judgments made by evaluators who have not been involved in problem detection introduces uncertainty regarding the interpretation of the problem reports. In collecting severity judgments one always has to balance the risk of biased severity judgments against that of misinterpreted problem reports.

It should be noted that our definitions of “severe” problems are not empirically founded, that is, we have little evidence that these problems would indeed be more problematic than the other problems if users were to encounter them in the real world. Certainly, the first method of identification (by problem detection criteria) has some empirical support (i.e., that at least one evaluator saw evidence in at least one user’s behavior of excessive delay, giving up, or a system crash), but all three methods require additional research to establish their validity.

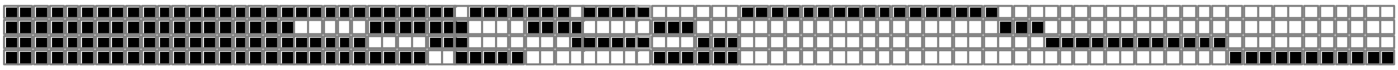

Figure 2. Matrix showing who found which problems. Each row represents an evaluator, each column a problem, and each black square that the evaluator detected the problem.

The substantial differences among the evaluators in terms of their selection of problems for their top-10 lists reveal that judgments of severity are highly personal. In fact, we were surprised that *no* UPT appear on *all* evaluators' top-10 lists. To further investigate the evaluators’ strategies in selecting severe problems we extended our study by collecting retrospective reports.

RETROSPECTIVE REPORTS

Though caution should be exercised in using retrospective reports the evaluators were asked to write down their strategy for creating their top-10 lists. Moreover, for each of the 15 UPTs appearing on a different evaluator’s top-10 list, but not on their own, they were asked to explain why they judged this UPT to be less severe than those on their own list.

E4 based her severity judgment solely on the frequency information on the master list. She first selected the UPTs experienced by all four users (5 UPTs). She then looked at the UPTs that were experienced by either two or three users but were detected by all evaluators, picking 5 of 10 possible UPTs. She did not read the content of all 93 UPTs. Filtering by frequency of occurrence and then by frequency of detection, she missed identifying a system crash as a severe UPT because it was experienced by only one user.

In contrast, E1, E2, and E3 read through all 93 UPTs highlighting potentially severe problems. The first pass reduced each evaluator's set of problems to between 15 and 25 UPTs. These three evaluators then read through their reduced set of problems removing UPTs gradually until they reached 10 UPTs.

Although they all used this ‘homing-in approach’ to selecting top-10 UPTs, the details of their approach differed. E1 reported focusing on the needs of experienced users in real-life situations. Six of her top-10 UPTs were explicitly selected for this reason and the same reason was given for excluding 10 of the 15 UPTs the other evaluators chose.

E2 used her opinion of the severity of the problem criteria in many of her decisions (6 of her top-10). The user frequency also played a role in E2’s creation of her top-10 list. She explicitly favored general problems over specific ones and tried to balance the needs of novice and experienced users. She rejected evidence that new features were needed (e.g., a *search* command) in favor of evidence that there were problems with existing features.

After the initial pass, E3 removed potentially severe UPTs by comparing them pair-wise, rather than relying on general heuristics. She based these comparisons on her ability to provide sound arguments for one or the other UPT, with no dominant pattern to these arguments. She described her last few decisions as “more or less random”, as the compared problems appeared almost equally severe.

In summary, the evaluators' methods for extracting top-10 UPTs varied greatly, according to both concurrent explanations for selecting problems and their retrospective reports. The selection methods were based on multiple aspects such as the evaluators’ favor for certain user groups, the number of evaluators and users encountering a problem, the violated problem criteria, expectations about real-world usage of the Builder, etc. All these aspects may catch important dimensions of problem severity but they also point out that severity is an ill-defined concept.

SUMMARY AND CONCLUSION

Our study shows that analyzing usability test sessions is an activity subject to considerable individual variability. When four research evaluators with extensive knowledge in HCI evaluated the same four usability test sessions, almost half of the problems were detected by only a single evaluator, while just 20% of the problems were detected by all evaluators.

The evaluators’ detection rate was higher for more severe problems. Severe problems were identified both by problem-detection criteria or inclusion on evaluators’ top-10 lists. However, all the sets of severe problems still displayed a substantial evaluator effect. Moreover, the evaluators disagreed substantially in their judgment of what constituted the ten most severe problems. None of the 25 problems in the union of the evaluators’ top-10 lists was selected as severe by all evaluators, and 56% appeared on only a single top-10 list.

The evaluator effect revealed in this study shows that usability tests are less reliable than previously reported. No single evaluator will detect all problems in an interface when analyzing usability test sessions, and any pair of evaluators is far from identifying the same set of severe problems.

FUTURE WORK

Clearly, this small study should be followed by larger studies examining how the evaluator effect manifests itself with such variables as different definitions of severity, instructions to the evaluators, problem detection criteria, evaluator training, system type, and task types. Our investigation of the evaluator effect asks many more questions than we can answer at this time.

ACKNOWLEDGMENTS

This work was supported by The Danish Research Councils and by the United States' National Science Foundation, Award# IRI-9457628. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of The Danish Research Council, the Danish Government, NSF, or the U. S. Government. We would like to thank Ejlif E. Jacobsen for his help with the statistical analysis.

REFERENCES

Bailey, R.W., Allan, R.W., & Raiello, P. (1992). Usability testing vs. heuristic evaluation: A head-to-head comparison. In Proceedings of the Human Factors Society 36th Annual Meeting. Santa Monica: HFS, 409-413.

Cuomo, D.L., & Bowen, C.D. (1994). Understanding usability issues addressed by three user-system interface evaluation techniques. Interacting with Computers, 6(1), 86-108.

Hackman, G.S., & Biers, D.W. (1992). Team usability testing: Are two heads better than one? In Proceedings of the Human Factors Society 36th Annual Meeting. Santa Monica: HFS, 1205-1209.

Held, J.E., & Biers, D.W. (1992). Software usability testing: Do evaluator intervention and task structure make any difference? In Proceedings of the Human Factors Society 36th Annual Meeting. Santa Monica: HFS, 1215-1219.

Henderson, R., Podd, J., Smith, M., & Varala-Alvarez, H. (1995). An examination of four user-based software evaluation methods. Interacting with Computers, 7(4), 412-432.

Holleran, P.A. (1991). A methodological note on pitfalls in usability testing. Behaviour & Information Technology, 10(5), 345-357.

Jacobsen, N.E., Hertzum, M., & John, B.E. (1998). The evaluator effect in usability tests. In ACM CHI'98 Conference Summary. Reading, MA: Addison-Wesley, 255-256.

Jeffries, R. (1994). Usability problem reports: Helping evaluators communicate effectively with developers. In J. Nielsen & R.L. Mack (Eds.), Usability Inspection Methods. New York: Wiley, 273-294.

John, B.E., & Marks, S.J. (1997). Tracking the effectiveness of usability evaluation methods. Behaviour & Information Technology, 16(4/5), 188-202.

John, B.E., & Mashyna, M.M. (1997). Evaluating a multimedia authoring tool. Journal of the American Society for Information Science, 48(11), 1004-1022.

Jørgensen, A.H. (1989). Using the thinking-aloud method in system development. In G. Salvendy & M.J. Smith (Eds.), Designing and Using Human-Computer Interfaces and Knowledge Based Systems. Amsterdam: Elsevier Science Publishers, 743-750.

Karat, C. (1994). A comparison of user interface evaluation methods. In J. Nielsen & R.L. Mack (Eds.), Usability Inspection Methods. New York: Wiley, 203-233.

Karat, C.-M., Campbell, R., & Fiegel, T. (1992). Comparison of empirical testing and walkthrough methods in user interface evaluation. In Proceedings of the ACM CHI'92 Conference. Reading, MA: Addison-Wesley, 397-404.

Lewis, J.R. (1994). Sample sizes for usability studies: Additional considerations. Human Factors, 36(2), 368-378.

Nielsen, J. (1993). Usability Engineering. Boston: Academic Press.

Nielsen, J., & Landauer, T.K. (1993) A mathematical model of the finding of usability problems. In S. Ashlund, K. Mullet, A. Henderson, E. Hollnagel, & T. White (Eds.), Proceedings of the InterCHI'93 Conference. New York: ACM, 206-213.

Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. In Proceedings of the ACM CHI'90 Conference. Reading, MA: Addison-Wesley, 249-256.

Ohnemus, K.R., & Biers, D.W. (1993). Retrospective versus concurrent thinking-out-loud in usability testing. In *Proceedings of the Human Factors and Ergonomics Society 37th Annual Meeting*, 2. Santa Monica: HFES, 1127-1131.

Pane, J.F., & Miller, P.L (1993). The ACSE multimedia science learning environment. In T.-W. Chan (Ed.), Proceedings of the 1993 International Conference on Computers in Education, (Taipei, Taiwan, December), 168-173.

Virzi, R.A. (1992). Refining the test phase of usability evaluation: How many subjects is enough? Human Factors, 34(4), 457-468.

Virzi, R. A., Sorce, J. F., & Herbert, L. B. (1993) A comparison of three usability evaluation methods: Heuristic, think-aloud, and performance-testing. In *Proceedings of the Human Factors and Ergonomic Society 37th Annual Meeting*. 1. Santa Monica, HFES, 309-313.