Available online at www.sciencedirect.com
ScienceDirect
Computers & Education 49 (2007) 441–459
COMPUTERS & EDUCATION
www.elsevier.com/locate/compedu

The role of errors in learning computer software

Robin H. Kay *

University of Ontario Institute of Technology, 2000 Simcoe St. North, Oshawa, Ont., Canada L1H 7K4

Received 28 July 2005; received in revised form 24 August 2005; accepted 16 September 2005

Abstract

Little research has been done examining the role of errors in learning computer software. It is argued, though, that understanding the errors that people make while learning new software is important to improving instruction. The purpose of the current study was to (a) develop a meaningful and practical system for classifying computer software errors, (b) determine the relative effect of specific error types on learning, and (c) examine the impact of computer ability on error behaviour. Thirty-six adults (18 males, 18 females), representing three computer ability levels (beginner, intermediate, and advanced), volunteered to think out loud while they learned the rudimentary steps (moving the cursor, using a menu, entering data) required to use a spreadsheet software package. Classifying errors according to six basic categories (action, orientation, knowledge processing, seeking information, state, and style) proved to be useful. Errors related to knowledge processing, seeking information, and actions were observed most frequently, however, state, style, and orientation errors had the largest immediate negative impact on learning. A more detailed analysis revealed that subjects were most vulnerable when observing, trying to remember, and building mental models. The effect of errors was partially related to computer ability, however beginner, intermediate and advanced users were remarkably similar with respect to the prevalence of errors.
© 2005 Elsevier Ltd. Allrights reserved.

Keywords: Computer error learn software classification cognitive expert novice

* Tel.: +1 905 721 3111 2679.
E-mail address: robin.kay@uoit.ca

0360-1315/$ - see front matter © 2005 Elsevier Ltd. Allrights reserved.
doi:10.1016/j.compedu.2005.09.006

1. Overview

Human error is inevitable, even when straightforward tasks are performed by experienced users (Hollnagel, 1993; Lazonder & Van Der Meij, 1995; Virvou, 1999). While extensive research has been done on the role of errors in high-risk domains, substantially less effort has been made in the area of computer software. The classification rubrics for high-risk domains do not translate well to a computer-based environment. Furthermore, most research in the computer software domain has looked at human–computer interaction (HCI) with a focus on improving software interfaces (Carroll, 1990; Hourizi & Johnson, 2001; Maxion, 2005; Norman & Draper, 1986; Reason, 1990). More research is needed on the role of errors in the learning process (Brown & Patterson, 2001; Reason, 1990).

The purpose of this paper was to (a) develop a meaningful and practical system for classifying errors made while learning a new computer software package, (b) explore the relative effect of specific error types on learning performance, and (c) examine the impact of computer ability on error behaviour.

2. Literature review

2.1. General research on errors

Extensive research has been done on identifying and evaluating the impact of errors in a wide variety of domains including air traffic control (Isaac, Shorrock, & Kirwan, 2002), nuclear power plants (Kim, Jung, & Ha, 2004), medicine (Horns & Lopper, 2002), aeronautics (Hourizi & Johnson, 2001), ATM machines (Byrne & Bovair, 1997), general safety systems (Vaurio, 2001), and telephone operation (Gray, John, & Atwood, 1993). Typically, these domains are high risk areas where making errors can result in serious loss of time, money or life. The principal goal of research, then, is to identify, predict and ultimately eliminate errors (Johnson, 1999). However, there is considerable evidence to suggest that all humans make errors, even experts (e.g., Kitajima & Polson, 1995; Norman, 1981; Reason, 1990) in the most straightforward of tasks (Brown & Patterson, 2001). In short, human error is inevitable (Hollnagel, 1993; Lazonder & Van Der Meij, 1995; Virvou, 1999).

2.2. Errors and human computer interaction

Research on errors in the domain of computers has focussed on system development (Johnson, 1999), software design (Smith & Harrison, 2002), operating systems (Brown & Patterson, 2001), computer supported co-operative work environments (Trepess & Stockman, 1999), programming (e.g., Ebrahim, 1994; Emurian, 2004), and HCI (e.g. Carroll, 1990; Norman & Draper, 1986). While errors in most of these domains (e.g., system and software design, operating systems, and programming) can result in considerable loss of time and money, errors in HCI usually present minimal risk. Making errors while learning a computer software package can be frustrating and personally time consuming, but is clearly less risky than a nuclear accident, an incorrect dosage of medicine, or a computer server shut down.

The relatively low-risk HCI milieu has implications on the kind of research undertaken. Errors are more readily accepted (Carroll, 1990; Lazonder & Van Der Meij, 1995; Norman & Draper, 1986), and the key focus of this research is to modify and improve user interfaces so that errors can be minimized (Carroll, 1990; Hourizi & Johnson, 2001; Maxion, 2005; Norman & Draper, 1986; Reason, 1990). The ultimate goal is to design error-free software that is easy to use for everyone (e.g., Carroll, 1990; Ebrahim, 1994; Norman & Draper, 1986).

Several researchers (e.g., Brown & Patterson, 2001; Kay, in press) have argued, though, that not enough emphasis is being placed on the human user and learning. Virvou (1999) and Rieman, Young, and Howes (1996) note that human reasoning is based on analogies, generalizations, and guessing when learning new ideas and procedures. These methods work reasonably well but are prone to errors particularly when a person is interacting with a computer – a machine that can only interpret precise instructions. Virvou (1999) and Rieman et al.'s (1996) claims are supported by observed error rates of 25–50% for novices (Lazonder & Van Der Meij, 1995) and 5–20% for experienced users (Card, Moran, & Newell, 1983; Norman, 1981; Reason, 1990). Finally, Brown and Patterson (2001) note that computer outages have remained virtually unchanged in the past three decades in spite of improvements in software interfaces and hardware. In summary, human error is not a problem that should be left solely to the user interface community (Brown & Patterson, 2001). There is a clear need for research examining the role of the human user in modifying and reducing errors.

2.3. Classification of errors

A considerable amount time and effort has been devoted to useful classification systems of errors (Emurian, 2004; Hollnagel, 2000; Hourizi & Johnson, 2001; Kitajima & Polson, 1995; Lazonder & Van Der Meij, 1995; Reason, 1990; Virvou, 1999). Reason (1990) proposed very general errors types: slips or lapses, rule-based mistakes, and knowledge-based mistakes. Slips occur when a correct plan or action is executed incorrectly (e.g., typing mistake, dropping an object, tripping, mispronounced word) whereas a lapse is typically a memory error. Mistakes are based on incorrect plans or models. Rule-based mistakes occur when a user applies an incorrect set of rules to achieve an action. When a person's collection of rule-based, problem solving routines is exhausted, he/she is forced to slow, conscious model building and can be subject to developing incorrect representations of a problem. These are known as knowledge-based errors. While this classification system has proven to be useful, Reason (1990) acknowledges that "there is no universally agreed classification of human error, nor is there any one prospect. A taxonomy is usually made for a specific purpose, and no single schema is likely to satisfy all needs" (p. 10).

Hollnagel (1993) contends, though, that there are eight basic errors can be used to classify any incorrect action involving timing, duration, force, distance/speed, direction, wrong objects, and sequence. However, Hollnagel's classification rubric has been tested in a limited range of high-risk domains.

A more specialized or domain-specific approach to error classification is supported by a number studies offering unique errors types including input and test errors while programming (Emurian, 2004), fixation (De Keyser & Javaux, 1996) and automation surprise (Hourizi & Johnson, 2001) errors experienced by pilots, post completion errors when cards are left in ATM machines (Byrne & Bovair, 1997), social conflict errors in collaborating computer-based communities (Trepess & Stockman, 1999), shift work and medication errors in hospitals (Inoue & Koizumi, 2004), fatal errors for computer server operators (Virvou, 1999), and entanglements or combination errors committed by software users (Carroll, 1990). It would be difficult for a general model of error classification to capture these domain-specific errors. Furthermore, generalizing error categories might take away rich contextual information needed to address and rectify problem areas.

To date, no classification system of computer software errors has been developed, although HCI researchers have informally identified a number of different error types, such as fixation, slips, and mistakes (Norman & Draper, 1986), going too fast, reasoning on the basis of too little information, inappropriate use of prior knowledge, and combination errors or entanglements (Carroll, 1990). Perhaps the most significant error is the inability of a learner to observe or recognize his or her mistakes (Lazonder & Van Der Meij, 1995; Virvou, 1999; Yin, 2001).

2.4. Role of errors in learning

Very little research has been done on attempting to understand the role of errors in the learning process (Reason, 1990). Three conclusions, noted earlier, indicate that this kind of research, though, is important. First, errors are inevitable when humans are performing any task (Hollnagel, 1993; Lazonder & Van Der Meij, 1995; Virvou, 1999) and remarkably frequent (5–50% – Card et al., 1983; Lazonder & Van Der Meij, 1995; Norman, 1981; Reason, 1990) in a learning situation, particularly when it involves computers (Virvou, 1999). Second, the role of the human in the error process needs to be studied in more detail to complement the extensive research on computer interfaces (e.g., Brown & Patterson, 2001; Kay, in press). Third, domain-specific classification rubrics need to be developed with a focus on cognitive activity and computers.

Hollnagel (2000) offered four general learning or cognitive categories for errors: execution, interpretation, observations, and planning. While relatively untested, these categories offer a starting point with which to investigate the role of errors in learning. Additionally, Reason's (1990) rule and knowledge-based error categories might be useful given the procedural and model building activities involved in learning computer software.

After identifying and classifying errors made while learning new software, it is equally important to examine how users recover from errors. Novices, for example, have been reported to need extensive, context-specific information when an error has occurred (Lazonder & Van Der Meij, 1995; Yin, 2001). Experienced users, on the other hand, have an affinity for recovering from errors quickly (Kitajima & Polson, 1995). Regardless of ability level, being forced to divert attention to "error" interruptions is common when interacting with computer software and can cause immediate short-term memory loss (Oulasvirta & Saariluoma (2004)). As well, the adequate handling of errors depends on what the users do with respect to detection, diagnosis, and correction (Lazonder & Van Der Meij, 1995). Ultimately, understanding the role of errors in learning can be instrumental to guiding effective instruction (Carroll, 1990).

2.5. Effect of ability

It is reasonable to expect that one's previous ability using computer software will affect the prevalence and impact of errors made. Experts are expected to outperform beginners in new learning environments. In fact, expertise has been examined in a number of domains including chess (Charness, 1991), physics (Anzai, 1991), medicine (Patel & Groen, 1991), motor skills in sports and dance (Allard & Starkes, 1991), music (Sloboda, 1991), and literacy (Scardamalia & Bereiter, 1991). The typical expertise paradigm involves comparing experts with novices on a series of tasks that experts can do well and that novices have never tried (Ericsson & Smith, 1991). However, Reason (1990) notes "no matter how expert people are at coping with familiar problems, their performance will begin to approximate that of novices once their repertoire of rules has been exhausted by the demands of a novel situation" (p. 58). The nature of expertise in using computer software has not been examined in the literature, particularly with respect to experts attempting unfamiliar tasks.

2.6. Purpose of study

The purpose of this study was threefold. First, a formative, post-hoc analysis was done to develop a meaningful and practical system for classifying errors specific to learning a new computer software package. Second, the relative effect of each error category on learning performance was examined. Finally, the impact of computer ability on error behaviour was evaluated.

3. Method

3.1. Sample

The sample consisted of 36 adults (18 males, 18 females): 12 beginners, 12 intermediates, and 12 advanced users, ranging in age from 23 to 49 ($M = 33.0$ years), living in the greater metropolitan Toronto area. Subjects were selected on the basis of convenience. Equal numbers of males and females participated in each ability group. Sixteen of the subjects had obtained their Bachelor's degree, eighteen their Master's degree, one a Doctoral degree, and one, a community college diploma. Sixty-four percent ($n = 23$) of the sample were professionals; the remaining 36% were students ($n = 13$). Seventy-two percent ($n = 26$) of the subjects said they were regular users of computers. All subjects voluntarily participated in the study.

3.2. Procedure

Overview. Each subject was given an ethical review form, computerized survey and interview before attempting the main task of learning the spreadsheet software package. Note that the survey and interview data were used to determine computer ability level. Once instructed on how to proceed, the subject was asked to think-aloud while learning the spreadsheet software for a period of 55 min. All activities were videotaped with the camera focused on the screen. Following the main task, a post-task interview was conducted.

Learning tasks. Spreadsheet software is used to create, manipulate and present rows and columns of data. The mean pre-task score for spreadsheet skills was 13.1 ($SD = 15.3$) out of a total possible score of 44. Ten of the subjects (6 advanced users, 4 intermediates) reported scores of 30 or more. None of the subjects had ever used the specific spreadsheet software package used in this study (Lotus 1-2-3).

Subjects attempted a maximum of five spreadsheet activities arranged in ascending level of difficulty including (1) moving around the spreadsheet (screen), (2) using the command menu, (3) entering data, (4) deleting, copying, and moving data, and (5) editing. They were first asked to learn "in general" how to do activity one, namely moving around the spreadsheet. When they were confident that they had learned this activity, they were then asked to complete a series of specific tasks. All general and specific activities were done in the order presented in Appendix A. In other words, subjects could not pick and choose what they wanted to learn.

From an initial pilot study of 10 subjects, it was determined that 50–60 min was a reasonable amount of time for subjects with a wide range of abilities to demonstrate their ability to learn the spreadsheet software package. Shorter time periods limited the range of activities that beginners and intermediate subjects could complete.

In the 55-min time period allotted to learn the software in the current study, a majority of the subjects completed all learning tasks with respect to moving around the screen (100%) and using the command menu (78%). About two-thirds of the subjects attempted to enter data (69%), although only one-third finished (33%) all the activities in this area. Less than 15% of all subjects completed the final tasks: deleting, copying, moving, and editing data.

3.3. Data collection

Think-aloud protocols. The main focus of this study was to examine the role of errors with respect to learning computer software. The use of think-aloud protocols (TAPs), where subjects verbalize what comes to their mind as they are doing a task, is one promising technique for examining transfer. Essentially, the think-aloud procedure offers a window into the internal talk of a subject while he/she is learning. Ericsson and Simon (1980), in a detailed critique of TAPs, conclude that "verbal reports, elicited with care and interpreted with full understanding of the circumstances under which they were obtained, are a valuable and thoroughly reliable source of information about cognitive processes" (p. 247).

The analyses used in this study are based on think-aloud data. Specifically, 627 learning behaviours involving errors were classified and rated according to the degree to which they influenced the learning.

Presentation of TAPs. The following steps were carried out in the think-aloud procedure to ensure high quality data:

Step 1. (Instructions) Subjects were asked to say everything they were thinking while working on the software. Subjects were told not to plan what they were going to say.
Step 2. (Examples) Examples of thinking aloud were given, but no practice sessions were done.
Step 3. (Prompt) Subjects were told it was important that they keep talking and that if they were silent for more than 5 s, they would be reminded to "Keep talking".
Step 4. (Reading) Subjects were permitted to read silently, but they had to indicate what they were reading and summarize when they had finished.
Step 5. (Giving help) If a subject was really stuck, he/she could ask for help. A minor, medium, or major form of help would be given, depending on how "stuck" a subject was.
Step 6. (Recording of TAPs) Both thinking aloud and the computer screen were recorded using an 8mm video camera.

3.4. Data source

Independent variables. There were six principal independent variables in this study corresponding to the six categories of errors made by subjects in this study. Note that errors were labeled according to what learning activity a subject was doing and included errors made when subjects were (a) actively doing something (action), (b) trying to find their current location or state of progress (orientation), (c) manipulating or processing information in some way (knowledge processing), (d) seeking information, (e) fixated or committing multiple errors simultaneously (in a state), and (f) acting in a unique fashion (style). Operational definitions for these six classifications and their respective sub-categories are presented in Table 1. Note that this classification system was driven by empirical observation, not theory.

In addition, three computer ability levels were compared in this study: beginners, intermediates, and advanced users. The criteria used to determined these levels included years of experience, previous collaboration, previous learning, software experience, number of application software packages used, number of programming languages/operating systems known, and application software and programming languages known. A multivariate analysis showed that beginners had significantly lower scores than those of intermediate and advanced users ($p < .005$), and intermediates users had significantly lower scores than advanced users on all eight measures ($p < .005$).

Dependent variables. The effect of each error category was evaluated using five dependent variables: how often an error was committed (frequency), influence the error had on learning (a score from -3 to 0 – see Table 2 for rating criteria), percentage of subjects who made an error, total error effect score, and total amount learned. Conceptually, the first three variables assessed prevalence (how often the behavior was observed and by how many subjects) and intensity (mean influence of leaning behaviour). The fourth variable, total error effect score, was a composite of the first three variables and was calculated by multiplying the frequency in which an error occurred by the mean influence score of the error by the percentage of subjects who made the error. For example, knowledge processing errors were made 154 times, had a mean influence of -1.56 and were made by 97% of the subjects. The total error effect score, then, was -233.0 ($154 \times -1.56 \times .97$).

Total amount learned was calculated by adding up the number of subgoal scores that each subject attained during the 55-min time period. For each task, a set of learning subgoals was rated according to difficulty and usefulness. For example, the task of "moving around the screen" had five possible subgoals that could be attained by a subject: using the cursor key (1 point), using the page keys (1 point), using the tab keys (1 point), using the GOTO key (2 points), and using the End-Home keys (2 points). If a subject met each of these subgoals successfully, a score of 7 would be given. If a subject missed the last subgoal (using the GOTO key), a score of 5 would be assigned.

Reliability of TAPs. Reliability and validity assessments were derived from the feedback given during the study and a post-task interview. One principle concern was whether the TAPs influenced learning. While, several subjects reported that the think-aloud procedure was "weird", "frustrating" or "difficult to do", the vast majority found the process relatively unobtrusive. Almost 70% of the subjects ($n = 25$) felt that thinking aloud had little or no effect on their learning.

The accurate rating of the influence of an error on learning (Table 2) was critical to the reliability and validity of this study. Because of the importance of the learning influence scores, six out-

Table 1
Operational definitions of errors

Error type	Criteria
Action error	
Observation	Does not observe consequences when key is pressed
Sequence	Types in/selects information in the wrong order
Syntax	Correct idea but types in incorrect syntax
Wrong key	Presses the wrong key
Orientation error	
General	Does not know where he/she is the program
Knowledge processing error	
Arbitrary connection	Makes arbitrary connection between two events
Missed connection	Misses connection between two events
Mistaken assumption	Makes mistaken assumption
Mental model	Misunderstanding in subject's mental model of how something worked
Over extension	Extends concept to an area in which it does not apply
Wrong search space	Subject choose wrong location in which to look for information
Too specific in focus	Subject focus is too narrow or specific
Misunderstands task	Subject misunderstands task in study
Terminology	Does not understand meaning of word or phrase
Seeking information error	
Attention	Shifts attention away from current task
Memory error	Forgets information that has been presented/read previously
Observe	Misreads or does not see a cue or piece of information
State error	
Combination	Combination of 2 or more error type
Fixation	(a) Repeats exact same activity at least three times when it is clear each time the activity does not work (b) Repeated activity occurs for more than 5 min with no progress made toward a solution
Style error	
Miscellaneous style	(e.g., random typing or turning of pages, taking the long safe route, stalling for time)
Pace	Doing an activity at a pace in which they miss information being presented
Premature closure	Believes he/she is finished task when there is more to complete

side raters were used to assess a 10%, stratified, random sample of the total 627 occasions when errors were made. Inter-rater agreement was calculated using Cohen' κ (Cohen, 1960), a more conservative and robust measure of inter rater agreement (Bakeman, 2000; Dewey, 1983). The κ coefficients for inter rater agreement between the experimenter and six external raters (within one point) were as follows: *Rater 1*, 0.80; *Rater 2*, 0.82; *Rater 3*, 0.95; *Rater 4*, 0.94; *Rater 5*, 0.93; *Rater 6*, 0.93. Coefficients of 0.90 or greater are nearly always acceptable and 0.80 or greater is acceptable in most situations, particularly for the more conservative Cohen's κ (Lombard, Snyder-Duch, & Bracken, 2004).

Table 2
Rating system for influence score

Score	Criteria used	Example
−3	A significant misunderstanding or mistake is evident that is judged to use a significant amount of time	Subject thinks that the software help is the main menu and spends 15 min learning to do the wrong task
−2	A significant misunderstanding or mistake which leads subject away from solving the task at hand	Subject believes all commands are on the screen and does not understand that there are sub menus. This results some time loss and confusion
−1	Minor misconception that has little effect on the direct learning of the task at hand	Subject tries HOME key, which takes him back in the wrong direction, but does not cause a big problem in terms of moving to the specified cell
0	(a) Activity has no apparent effect on progress OR (b) Can't directly determine effect of activity OR (c) Both good and bad effects	(a) Subject tries a key and it does not work (e.g., gets beeping sound) (b) Subject gets upset, but it is hard to know how it affects future actions (c) Subject moves to cell quickly, but fails to learn a better method. It is good that he completed the task, but bad that he did not learn a more efficient method

4. Results

4.1. Frequency of errors made

The average number of errors per subject for the 55-min learning period was 17.4 ($SD = 9.4$). Errors were experienced most often when subjects were seeking information ($n = 170$), processing knowledge ($n = 154$), or carrying out some action ($n = 131$). The most frequent subcategory errors occurred when subjects were observing either their own actions or while seeking information ($n = 161$), trying to remember information ($n = 93$), attempting to create a mental model ($n = 84$), or a committing a combination of errors ($n = 54$). The frequency of each error category is presented in Table 3.

4.2. Mean influence of errors on learning

There was a significant difference among the six main error categories with respect to their immediate influence on learning ($p < .001$; Table 4). State ($M = -1.90$), orientation ($M = -1.73$), and knowledge processing errors ($M = -1.56$) were significantly more detrimental than seeking information ($M = -1.17$) and action errors ($M = -1.10$) (Scheffé post hoc analysis; $p < .005$; Table 4).

Subcategories with a mean influence of −1.60 or less included mental model ($M = -1.67$), wrong search space ($M = -1.75$), terminology ($M = -1.64$), combination ($M = -2.00$), and pace ($M = -1.67$) errors. A statistical comparison among subcategories of errors could not be done because of the small sample size (Table 5).

Table 3
Frequency of errors made

Error type	Frequency	% of all errors
Seek information		
Memory	93	15
Observe	72	11
Attention	5	1
Total	170	27
Knowledge processing		
Mental models	84	13
Mistaken assumption	22	4
Terminology	11	2
Over extension	9	1
Wrong search space	8	1
Misunderstands task	6	1
Too specific in focus	5	1
Arbitrary connection	5	1
Missed connection	4	1
Total	154	25
Action		
Observation	89	14
Wrong key	28	4
Syntax	7	1
Sequence	7	1
Total	131	21
State		
Combination	54	9
Fixation	14	2
Total	68	11
Style		
Premature closure	30	5
Pace error	15	2
Misc. style	10	2
Total	55	9
Orientation		
Total	49	8

Table 4
Analysis of variance for error type as a function of mean influence on learning score

Source	Sum of squares	df	Mean square	F
Between groups	47.24	5	9.45	14.00*
Within groups	419.28	621	0.68	
Total	466.52	626		

* $p < 0.001$.

Table 5
Total error effect as a function of error type

Error type	Count	% of subjects	Mean influence (*SD*)	Total error effect[a]
Knowledge processing				
Mental model	84	78	−1.67 (0.8)	−109.4
Mistaken assumption	22	47	−1.36 (0.6)	−14.1
Terminology	11	28	−1.64 (0.7)	−5.1
Wrong search space	8	19	−1.75 (0.9)	−2.7
Over extension	9	17	−1.33 (1.0)	−2.0
Misunderstands task	6	14	−1.33 (0.8)	−1.1
Too specific in focus	5	14	−1.20 (0.8)	−0.8
Arbitrary connection	5	11	−1.40 (0.6)	−0.8
Missed connection	4	11	−1.25 (1.0)	−0.6
Total	154	97	−1.56 (0.8)	−233.0
Seek information				
Memory	93	89	−1.00 (0.7)	−82.8
Observe	72	78	−1.39 (0.9)	−78.1
Attention	5	14	−1.20 (0.8)	−0.8
Total	170	97	−1.17 (0.8)	−192.9
Action				
Observation	89	86	−1.40 (0.7)	−107.2
Wrong key	28	58	−0.43 (0.4)	−7.0
Syntax	7	8	−0.71 (0.7)	−0.4
Sequence	7	11	−0.29 (1.1)	−0.2
Total	131	94	−1.10 (0.9)	−135.5
State				
Combination	54	53	−2.00 (0.8)	−57.2
Fixation	14	28	−1.50 (0.8)	−5.9
Total	68	67	−1.90 (0.8)	−86.6
Style				
Premature closure	30	61	−1.33 (0.6)	−24.3
Pace error	15	28	−1.67 (0.6)	−7.0
Misc. style	10	17	−1.60 (0.5)	−2.7
Total	55	75	−1.47 (0.6)	−60.6
Orientation				
Total	49	53	−1.73 (1.0)	−44.9
All errors				
Total	627	100	−1.40 (0.9)	−877.8

[a] Calculated by multiplying frequency by % of subjects who made this error type by mean influence on learning.

4.3. Percentage of subjects who made errors

It is clear from Table 5 that all subjects, regardless of ability, made errors while learning. Knowledge processing, seeking information, and action errors were made by over 90% of all subjects. State (75%) and style (67%) were observed less often, and only half the subjects experienced orientations errors.

With respect to specific subcategories, memory errors (89%), failing to accurately observe the consequences of one's actions (78%), inaccurate mental models (78%), and observation errors while seeking information were experienced by a majority of the subjects.

4.4. Total error effect score

Knowledge processing, seeking information, and action errors showed the highest total error effect scores, largely because these kinds of errors were made frequently by almost all subjects (Table 5). State, style, and orientation errors showed relatively low total error effect scores because they were experienced less often by fewer subjects.

4.5. Total amount learned

Only one of the six main categories, orientation errors, showed a significant correlation ($r = -0.57$, $p < .05$) with total amount learned. This result is consistent with the relatively high mean influence on learning score observed for orientation errors, but not with the low total effect score.

4.6. Computer ability level and errors made

Frequency of errors. There were no significant differences among beginner ($M = 20.4$; $SD = 11.7$), intermediate ($M = 16.8$; $SD = 10.0$) and advanced ($M = 15.1$; $SD = 5.6$) groups with respect to the number of errors made.

Mean influence on learning. A two-way ANOVA revealed significant differences among ability levels ($p < .001$), but no interaction effect between error type and ability level (Table 6). Advanced users ($M = -1.15$; $SD = 0.86$) were affected by errors significantly less than either intermediate ($M = -1.40$; $SD = 0.81$) or beginner users ($M = -1.58$ $SD = 0.86$) (Scheffé post hoc analysis, $p < .05$).

Orientation errors. While advanced users were clearly less effected by errors than intermediate or beginners (Table 6), a closer examination of frequency of errors, percentage of subjects who made errors, and mean influence of errors on learning as a whole revealed notable similarities among all three ability groups with one exception: orientation errors. Advanced users committed this kind of error infrequently and recovered quickly (Table 7). This turned out to be a significant

Table 6
Two-way analysis of variance for error type and ability as a function of mean influence on learning score

Source	Sum of squares	df	Mean square	F
Ability	11.27	2	5.63	8.52*
Error category	32.76	5	6.55	9.90*
Ability * Error category	4.70	10	0.47	0.71
Within cells	402.7	609	0.66	
Total	466.52	626		

* $p < 0.001$.

Table 7
Frequency, percent of subjects who made error, and mean influence on learning score as a function of ability level

Error category	Frequency			Percent of subjects who made error			Mean influence on learning		
	B[a]	I[a]	A[a]	B	I	A	B	I	A
Actions	45	45	41	100	83	100	−1.24	−1.15	−0.88
Orientation	32	14	3	67	67	25	−1.88	−1.57	−1.00
Know Processing	67	43	44	100	92	100	−1.64	−1.53	−1.45
Seeking Info	51	64	55	92	100	100	−1.27	−1.23	−1.00
State	33	21	14	75	75	75	−2.09	−2.00	−1.29
Style	17	14	24	75	75	75	−1.64	−1.42	−1.37

[a] B, beginner; I, intermediate; A, advanced.

advantage as "orientation errors" was the only category that was significantly and negatively correlated with total amount learned.

5. Discussion

5.1. Classification system for computer software domain

The three error categories (slips/lapses, rule-based mistakes, and knowledge-based mistakes) proposed by Reason (1990) can be applied to a number of the error types identified in this study. Pressing the wrong key, typing in an incorrect command, and forgetting newly learned information fits into the slips/lapses category. Selecting the wrong sequence of actions, making a mistaken assumption, and over extending a strategy aligns reasonably well with rule-based errors. Finally, having an incorrect mental model, misunderstanding a task, not understanding terminology, and making arbitrary connections appear to be knowledge-based errors. However, using Reason's (1990) more general categories, while parsimonious, eliminates key contextual clues about the circumstances surrounding error behaviour. In addition, combination, fixation, orientation, and style errors have no obvious place in Reason's classification rubric.

Hollnagel's (2000) cognitive error classification system (execution, interpretation, observations, and planning) is also a reasonable model for the errors observed in this paper. There appears to be a good fit for action and execution errors, knowledge processing and interpretation errors, and seeking information and observation errors. However, Hollnagel's (1993) planning category does not match the typical tasks performed by someone learning a new software package. Deliberate, well-thought out actions appear to be the exception (see Kay, in press). Virvou's (1999) approximate reasoning, trial and error, guessing paradigm is a closer match to what occurred in this study. It is worth noting that Hollnagel's error model, like Reason's (1990) model, would eliminate useful descriptive details. As well, the model fails to account for domain-specific errors like fixation and combination mistakes.

In this study, errors were organized according to what subjects were doing in the learning process when they made their mistake. This richer, purpose-focused, classification system provides (a) a better understanding of the knowledge building process, and (b) specific opportunities for improving instruction. This system also proved to be consistent with errors informally observed in HCI research (e.g., Carroll, 1990; Norman & Draper, 1986): fixation, going too fast, reasoning on the basis of too little information, inappropriate use of prior knowledge, and combination errors or entanglements (Carroll, 1990).

6. Effect of errors on learning

The findings from this study suggest that all subjects, regardless of ability level, make errors throughout the entire computer knowledge acquisition process: when they look for useful information, when they observe the result of their keystrokes, when they attempt to develop a model to understand what they have learned, and when they make judgements about whether they have achieved their final goal. This result is consistent with claims of error inevitability (Hollnagel, 1993; Lazonder & Van Der Meij, 1995; Virvou, 1999).

The most frequent errors experienced by over 90% of all subjects were those related to seeking information, knowledge processing, and interacting with the software. More specifically, subjects appear most vulnerable with respect to observation, memory, and model building errors. These weak spots are indirectly supported by previous research. Lazonder and Van Der Meij (1995) noted that knowing when a mistake occurs and its exact nature can be vital to success. If a subject fails to observe what has happened (observation error), learning can be severely limited. Oulasvirta and Saariluoma (2004) add that attending to interruptions, a typical state of affairs while learning computer software, can lead to short-term memory loss (memory error). Finally, because the software in this study was new to all subjects, Reason (1990) predicted that the probability of committing knowledge-based errors (model building error) would increase.

It is worthwhile to note that the most frequent errors were not the most detrimental to learning. State, style and orientation errors, which were observed relatively infrequently, had the highest negative mean influence on learning. In other words, specific errors types, even if they don't occur often, can appreciably interrupt the learning process. Virvou's (1999) "fatal" error category might be useful here. This kind of error is fatal in the sense that considerable time is lost while learning.

Orientation errors were noteworthy for two reasons. First they were the only error type significantly and negatively correlated with learning. Second, they appeared to affect beginner and intermediate users more than advanced users. This kind of error, however, has not been emphasized in previous HCI research (e.g., Carroll, 1990; Norman & Draper, 1986). More research needs to be done on how to address this kind of problem for new users.

For the most part, errors have an immediate negative effect on learning behavior, but are not significantly related to overall performance or total amount learned. This result may reflect the fact that errors are a natural component of learning, regardless of ability level, and that while they have an immediate negative effect, other learning behaviors (e.g., transfer knowledge – see Kay, in press) have a more significant and direct impact on overall learning performance.

6.1. Errors and computer ability

Previous expertise research suggests that advanced users would make fewer errors while learning and that the consequences of these errors would be less severe (e.g., Kitajima & Polson, 1995).

The latter conclusion was supported by this study, but not the former. The reason for this discrepancy may be due to the research paradigm used. In a typical expertise research design, experts are asked to do tasks they know quite well – little if any learning is required. In this study, advanced users were asked to learn software they had never used before. In a true learning situation, it appears that subjects of all ability levels make a full range of errors. This result is consistent with Reason's, 1990 proposition that more experienced users will start to look like novices when exposed to unfamiliar situations.

6.2. Suggestions for educators

An examination of the kinds of errors subjects make while learning suggests that help is needed in a variety of areas. Educators should be wary of the following problems:

(1) Careful observation of one's actions is critical for success.
(2) Errors due to forgetting or poor mental models were frequent. Assuring that new learners have adequate representations of computer-based concepts might be one way of helping them avoid making costly errors.
(3) Orientation errors, although relatively infrequent, need to be addressed because they are particularly influential on immediate and overall learning. Providing new users with clear cues about where they are and what they are doing at any given moment may be important, particularly for beginners and intermediates.
(4) Subjects, regardless of computer proficiency, will have more difficulty when they become fixated on a problem or when they experience more than one error at the same time.
(5) With the exception of orientation errors, expect subjects of all ability levels to experience a full range of errors.

6.3. Future research

This study is a preliminary first step into investigating the role of errors in learning a new software package. This research needs to be expanded in three key areas:

(1) test the classification scheme on a broader range of computer software;
(2) explore how users recover from errors; and
(3) evaluate various intervention strategies based on a well-developed error rubric.

6.4. Caveats

No research endeavor is without flaws. These factors should be considered when interpreting the results and conclusions of this study:

(1) Although over 600 learning activities were analyzed, the sample consisted of only 36 subjects, who were highly educated, and in their thirties. The results might be quite different for other populations.

(2) Only one software package was examined – spreadsheets. A variety of software packages need to be examined to increase the confidence in the results of this study.
(3) Procedural factors such as thinking-aloud and the presence of an experimenter did alter learning. Stress, for example, can increase errors rates significantly (Brown & Patterson, 2001).
(4) Subjects did not choose to learn this software for a personally significant reason. Reduced motivation may have affected error behaviour (Trepess & Stockman, 1999).
(5) The think-aloud process, while fairly comprehensive, captured only a subset of subjects' thoughts during the learning process. The classification system of errors, then, is compromised because one cannot truly know what is going on in the user's mind.

6.5. Summary

A six-category classification system of errors, based on a subject's purpose or intent while learning, was effective in identifying influential behaviors in the learning process. Errors related to knowledge processing, seeking information and actions were observed most frequently, however, state, style, and orientation errors had the largest immediate impact on learning. A more detailed analysis revealed that subjects were most vulnerable when observing, trying to remember, and building mental models. The effect of errors was partially related to computer ability, however beginner, intermediate and advanced users were remarkably similar with respect to the prevalence and impact of errors.

Appendix A. Specific spreadsheet tasks presented to subjects

General Task 1: Moving around the screen
Specific Tasks 1:

(a) Move the cursor to B5.
(b) Move the cursor to B161.
(c) Move the cursor to Z12.
(d) Move the cursor to A1.
(e) Move the cursor to HA1235.
(f) Move the cursor to the bottom left corner of the entire spreadsheet.

General Task 2: Using the command menu
Specific Tasks 2:

(a) Move to the command menu, then back to the worksheet.
(b) Move to the command: Save.
(c) Move to the command: Sort.
(d) Move to the command: Retrieve.
(e) Move to the command: Set Width.
(f) Move to the command: Currency.

General Task 3: Entering data into a cell

Specific Tasks 3:

NAME	TELEPHONE	SEX	DATE DUE	AMOUNT
Robin	900-0100	M	07/14/92	300.12
Mary	800-0200	F	06/16/92	20046.23

(a) Please start in cell A1 and enter all the information above.
(b) Centre the title SEX.
(c) Right justify the title AMOUNT.
(d) Widen the TELEPHONE column to 15 spaces.
(e) Narrow the SEX column to 5 spaces.

General Task 4: Deleting, copying, and moving data

Specific Tasks 4:

DATA A	DATA C	DATA B	DATA D
10	1	100	11
15	2	200	22
20	3	300	33
25	4	400	44
30	5	500	55

(a) In the table above, move everything in Column A to Column B.
(b) Delete Row 4.
(c) Delete Column A.
(d) Delete the numbers 300–500 in the DATA B column.
(e) Name the range of data in DATA C column. Call this range DATA C.
(f) Copy the underline under DATA 1 to the cells under DATA B, C and D.

General Task 5: Editing data

Specific Tasks 5:

```
Cana dian
Amrican
          70002
Mistake
Replace Me
```

(a) In the table above, delete the space in Cana dian.
(b) Add an “e” to Amrican.
(c) Change 70002 to 80002.
(d) Delete the word Mistake.
(e) Replace the phrase Replace Me with the phrase New Me.

References

Allard, F., & Starkes, J. L. (1991). Motor-skill experts in sports, dances, and other domains. In K. A. Ericsson & J. Smith (Eds.), *Toward a general theory of expertise* (pp. 126–152). Cambridge: Cambridge University Press.

Anzai, Y. (1991). Learning and use of representations for physics expertise. In K. A. Ericsson & J. Smith (Eds.), *Toward a general theory of expertise* (pp. 64–92). Cambridge: Cambridge University Press.

Bakeman, R. (2000). Behavioral observation and coding. In H. T. Reis & C. M. Judge (Eds.), *Handbook of research methods in social and personality psychology* (pp. 138–159). New York: Cambridge University Press.

Brown, A., & Patterson, D. A. (2001). To err is human. In *Proceedings of the first workshop on evaluating and architecting system dependability*, goeteborg, sweden.

Byrne, M. D., & Bovair, S. (1997). A working memory model of a common procedural error. *Cognitive Science, 21*(1), 31–61.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human–computer interaction.* Hillsdale: N.J.L. Erlbaum.

Carroll, J. B. (1990). *The Nurnberg funnel.* Cambridge, MA: MIT Press.

Charness, N. (1991). Expertise in chess: the balance between knowledge and search. In K. A. Ericsson & J. Smith (Eds.), *Toward a general theory of expertise* (pp. 39–63). Cambridge: Cambridge University Press.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*, 37–46 Winter.

De Keyser, V., & Javaux, D. (1996). Human factors in aeronautics. In F. Bodart & J. Vanderdonckt (Eds.), *Proceedings of the eurographics workshop, design, specification and verification of interactive systems'96*. Wien (Autriche): Springer-Verlag Computer.

Dewey, M. E. (1983). Coefficients of agreement. *British Journal of Psychiatry, 143*, 487–489.

Ebrahim, A. (1994). Novice programmer errors: language constructs and plan composition. *International Journal of Man–Machine Studies, 41*, 457–480.

Emurian, H. H. (2004). A programmed instruction tutoring system for Java: consideration of learning performance and software self-efficacy. *Computers in Human Behavior, 20*, 423–459.

Ericsson, A. K., & Smith, J. (1991). Prospects and limits of the empirical study of expertise: an introduction. In K. A. Ericsson & J. Smith (Eds.), *Toward a general theory of expertise* (pp. 1–38). Cambridge: Cambridge University Press.

Ericsson, K. A., & Simon, H. A. (1980). Verbal reports as data. *Psychological Review, 87*(3), 215–251.

Gray, W. D., John, B. E., & Atwood, M. E. (1993). Project Ernestine: validating a GOMS analysis for predicting and explaining real-world performance. *Human–Computer Interaction, 8*(3), 237–309.

Hollnagel, E. (1993). The phenotype of erroneous actions. *International Journal of Man–Machine Studies, 39*, 1–32.

Hollnagel, E. (2000). Looking for errors of omission and commission or the hunting of the snark revisited. *Reliability Engineering and System Safety, 68*, 135–145.

Horns, K. M., & Lopper, D. L. (2002). Medication errors: analysis not blame. *Journal of Obstetric, Gynecologic, and Neonatal Nursing, 31*, 355–364.

Hourizi, R., & Johnson, P. (2001). In Michitaka Hirose (Ed.), *Proceedings of INTERACT 2001, eighth IFIP TC.13 conference on human–computer interaction, Tokyo, July 9–14*. IOS Press.

Inoue, K., & Koizumi, A. (2004). Application of human reliability analysis to nursing errors in hospitals. *Risk Analysis, 24*(6), 1459–1473.

Isaac, A., Shorrock, S. T., & Kirwan, B. (2002). Human error in European air traffic management: The HERA project. *Reliability Engineering and System Safety, 75*, 257–272.

Johnson, C. (1999). Why human error modeling has failed to help systems development. *Interacting with Computers, 11*, 517–524.

Kay, R. H. (in press). Learning performance and computer software: an exploration of knowledge transfer. *Computers in Human Behavior*.

Kim, J. W., Jung, W., & Ha, J. (2004). AGAPE-ET: a methodology for human error analysis of emergency tasks. *Risk Analysis, 24*(5), 1261–1277.

Kitajima, M., & Polson, P. G. (1995). A comprehension-based model of correct performance and errors in skilled, display-based human–computer interaction. *International Journal of Computer Studies, 43*, 65–99.

Lazonder, A. W., & Van Der Meij, H. (1995). Error-information in tutorial documentation: supporting users' errors to facilitate initial skill learning. *International Journal of Computer Studies, 42*, 185–206.

Lombard, M., Snyder-Duch, J., & Bracken, C. C. (2004). Practical resources for assessing and reporting intercoder reliability in content analysis research projects. Retrieved September, 2004. Available from <http://www.temple.edu/mmc/reliability>.

Maxion, R. A. (2005). Improving user-interface dependability through mitigation of human error. *International Journal of Human–Computer Studies, 63*, 25–50.

Norman, D. A. (1981). Categorization of action slips. *Psychological Review, 88*, 1–15.

Norman, D. A., & Draper, S. W. (Eds.). (1986). *User centered system design: New perspectives on human–computer interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Oulasvirta, A., & Saariluoma, P. (2004). Long-term working memory and interrupting messages in human–computer interaction. *Behavior Information Technology, 23*(1), 53–64.

Patel, V. L., & Groen, G. J. (1991). The general and specific nature of medical expertise: a critical look. In K. A. Ericsson & J. Smith (Eds.), *Toward a general theory of expertise* (pp. 93–125). Cambridge, NY: Cambridge University Press.

Reason, J. (1990). *Human error*. New York, NY: Cambridge University Press.

Rieman, J., Young, R. M., & Howes, A. (1996). A dual-space model of iteratively deepening exploratory learning. *International Journal of Computer Studies, 44*, 743–775.

Scardamalia, M., & Bereiter, C. (1991). Literate expertise. In K. A. Ericsson & J. Smith (Eds.), *Toward a general theory of expertise* (pp. 172–194). Cambridge, NY: Cambridge University Press.

Sloboda, J. (1991). Musical expertise. In K. A. Ericsson & J. Smith (Eds.), *Toward a general theory of expertise* (pp. 153–171). Cambridge, NY: Cambridge University Press.

Smith, S. P., & Harrison, M. D. (2002). Blending descriptive and numeric analysis in human reliability design. In P. Forbrig, B. Urban, J. Vanderdonckt, & Q. Limbourg (Eds.), *Lecture notes in computer scienceInteractive systems: Design, specification and verification (DSVIS 2002)* (pp. 223–237). Springer.

Trepess, D., & Stockman, T. (1999). A classification and analysis of erroneous actions in computer supported co-operative work environment. *Interacting with Computers, 11*, 611–622.

Vaurio, J. K. (2001). Modelling and quantification of dependent repeatable human erros in system analysis and risk assessment. *Reliability Engineering and System Safety, 71*, 179–188.

Virvou, M. (1999). Automatic reasoning and help about human errors in using an operating system. *Interacting with Computers, 11*, 545–573.

Yin, L. R. (2001). Dynamic learning patterns: temporal characteristics demonstrated by the learner. *Journal of Educational Multimedia and Hypermedia, 10*(3), 273–284.