Usability Problems in Word Processing Applications

Michael S. Wogalter and Jennifer A. Cowley
North Carolina State University
Psychology Department, Human Factors and Ergonomics Program, Raleigh, NC 28695-7650 USA

Word processing is a major work activity for many occupations, including professionals and students. Although word processing applications have been available since the earliest personal computers, and some of these applications have undergone many versions and revisions, it might be expected that usability problems at this point in their evolution would be minimal. The present study suggests that there is still room for improvement. Specific findings include: (a) a preference for toolbars buttons with both icons and textual descriptors, (b) some automatic features such as "assistants" are disliked, and (c) people wanting easier ways to turn off the automatic features. Implications of these results are discussed with a focus on encouraging developers to make use of user data to enhance their product's usability. Such developments would benefit users' work productivity and satisfaction.

INTRODUCTION

It has been approximately three decades since computers, printers, and word processing software replaced typewriters. The newer tools are vastly better than the typewriter). Since then, people in various jobs and learning environments use word processing applications on a regular basis (e.g., Stainton-Skinn, 1982).

Rudimentary word processors have evolved together with computers over time, including the change from command based to graphical user interfaces (GUI). Microsoft Word, developed originally for the early GUI Apple Macintosh computer, has become a dominant word processing application across operating system (OS) platforms.

GUI platforms often utilize pull-down menus, which are beneficial because users can make use of people's excellent recognition capabilities as opposed to making use of people's poorer ability to recall information (e.g., Wiedenbeck & Davis, 1997). However, pull down menus can also be disadvantageous under some circumstances. In GUI interfaces, making repetitive commands and accessing deep hierarchical menus are generally more difficult, awkward and slower. While novice users generally prefer pull down menus, expert users have a greater preference for keyboard commands to accomplish tasks faster (e.g., Wogalter & Frei, 1990).

Another GUI method to deliver computer commands is through selection from toolbar menus. Toolbar menus are frequently comprised of set of several icons on a rollup ribbon, toolbar or a ruler. When turned on and displayed in an application, the icons can be selected by a single pointer click. The use of icons has sometimes been credited as one of the reasons why GUI interfaces are easier than command-line interfaces (e.g., Lodding, 1983). This simple attribution is probably not exactly true because there are numerous aspects of GUIs that aid usability besides icons. Additionally, a large growing body of research shows that icons (and other kinds of symbolic graphical forms) vary considerably in understandability (Wogalter, Silver, Leonard & Zaikina 2006). Whereas, some icons are comprehended with ease, others are not. On-screen icons are usually small and of relatively low resolution. They also may not represent some concepts clearly (James, Lynk, Molinarie, & Caird, 1995). A symbol that directly represents a concrete concept will generally be better understood than one that indirectly represents a lesser-known or abstract concept. As a result, people can fail to

comprehend the meaning of various kinds of symbolic forms (e.g., Leonard, Otani & Wogalter, 1999; Mehlenbacher, Duffy, & Palmer, 1989; Schwalm, Shaviv, & Goldschmidt, 2000). By extension to the computer interface, if users cannot identify or recognize the associated meaning of icons in a toolbar, then the interface is not as useful as it might otherwise be (Luximon, Ghambaryan, & Mehrabyan, 2003).

A potential way to raise icon understanding is to add textual definitions to the icons so that the symbol and textual definition are shown together. In other words, instead of the usual icons-only toolbar, the toolbar would contain words or abbreviations in addition to the icons. Indeed, some computer programs give descriptor words for icons in different ways. One method is to hover over the icon (tool tip), which after a short delay, textual descriptors are displayed in an adjacent area. Other programs give icon definitions using the hovering method but deliver the text descriptors at a different designated (constant) location on the screen. With both of these methods, each icon must be hovered over individually before its definition is shown and the process requires more time and some effort compared to a method in which the definitions always appear with the icons. Thus a potentially better method of displaying toolbar choices would be to have the descriptive text always present with the icons on the tool bar. One purpose of the present study was to evaluate people's judgments regarding having icons and textual descriptors in word processing toolbars.

Over the years, the number of features provided in word processing programs has increased, and consequently, the number of potential icons has increased. While the available space for icons has increased over the years due to the availability of larger screens, the amount of screen space allocated for toolbar icons will to some extent limit the area for the main document being worked on. The point here is that large areas used for toolbars and icons could adversely affect word processing performance because parts of the document are partially obscured (or a reduced, less legible document is used). Thus it is possible that combining icons with textual descriptions would not be desirable due usurpation of valuable screen space for the document itself.

The present study also examined people's beliefs about other word processor features beyond the toolbar. The aspects examined were based partly on the items used in the research of Levine and Wogalter (2000). They found that people strongly disliked certain features of word processing applications. In the present study, some of the aspects identified by Levine and Wogalter (2000) were examined. The categories of problem features examined concerned formatting, on-line help, and automation. These aspects were focused upon because participants in the earlier study indicated that some functions were needlessly complex, not obvious to use, and that turning off functions was difficult and that default automation was frequently disliked. Thus, as a follow-up to earlier Levine and Wogalter (2000) study, participants were asked which features gave them difficulty in using word processors. These data were collected to determine if there were consistently voiced problems.

Thus one purpose of this research was to examine whether participants would prefer textual descriptions added to tool bars. Another main purpose was to examine whether people had any other issues with the word processor(s) that they use. We believed that there are substantial usability problems to current word processor programs despite numerous iterations of revisions and versions over the past 30 years. On the other hand, if we were to find that participants provide ratings indicating that they experience only limited problems with their current word processing programs then this would suggest that the software has reached a level of maturity in terms of usability.

METHOD

Participants

A total of 187 individuals, with ages ranging from 18 to 79 years (M = 32.2, SD = 15.1) participated (47% male and 53% female). Ninety-eight were non-student adults and 89 were

undergraduates. Overall, race/ethnic composition was 83.3% Caucasian, 8.1% Asian, and 4.3% African/African American.

Materials and Procedure

Participants were given a multi-topic questionnaire. One section contained questions about computers including which word processors they have used, and the amount of time they use a computer and word processing software per week. Participants provided a self-report of their word processing expertise on a Likert-type 9-point scale with the following text and numeric anchors: (0) beginner, (4) intermediate and (8) advanced. They were then asked to evaluate a set of features of word processing software.

One set of questions had three items that described toolbars with respect to having icons and text descriptors. They were then asked to rate the importance of each item on a 9-point rating scale with the following anchors: (0) not at all important, (2) slightly important, (4) important, (6) very important, and (8) extremely important. The three items regarding the use of icons and text descriptors in toolbars were:

a. The toolbar should have only text descriptions for the different functions. Icons are not necessary.
b. The toolbar should only have icons for the different functions. Text descriptions are not necessary.
c. The toolbar should have both text descriptions and icons for the different functions.

In addition, several other features concerning formatting, on-line help, and automation were evaluated. Table 2 shows the specific list of items evaluated. The same 9-point Likert-type rating scale measuring judged importance, as described above, was used.

Participants were presented with one of two or orders of features to rate. One was randomized and the other was the reverse order of that.

Lastly, participants were asked to list difficulties that they have experienced with their current word processing program(s) that have caused difficulty, reduced their productivity, and that they would recommend being improved by software designers.

RESULTS

Participants reported spending an average of 26.57 (SD = 18.5) hours per week using the computer and 8.47 (SD = 10.3) hours per week using word processing software.

The data indicated that 88.2% of the participants reported using Microsoft Word, 2.1% Word Perfect and 1.6% Microsoft Works. Several other word processing applications were mentioned less frequently. On average, participants rated their level of expertise between intermediate and advanced according to the rating scale anchors (M = 5.27, SD = 1.8).

Table 1 shows the means and standard deviations for the ratings of toolbars with icons only, text only, or both icons and text. A one-way repeated measures ANOVA identified a significant effect, $F(2, 372) = 82.70$, $MSE = 4.73$, $p < 0.0001$. Comparisons among the means using Tukey's HSD test revealed that the "text/icon combination" (M = 5.54) was rated significantly higher than both the icon-only (M = 3.26) and the text-only (M = 2.85) mean ratings. A planned comparison indicated the icon-only toolbar was rated significant higher than the text-only condition, $F(1, 186) = 4.29$, MSE = 3.69, $p < 0.05$.

TABLE 1. *Mean importance ratings and standard deviations of word processing features on a 0 (low) to 8 (high) point scale.*

Toolbar Content	Mean	SD
Text only	2.85	2.3
Icons only	3.26	2.4
Both text and icons	5.53	2.2

Table 2 shows the mean ratings (and standard deviations) for several word processor features. The Tukey's HSD critical difference at $p < .05$ is .68, meaning that any difference between means bigger than this number in Table 2 is significant. Although the list of features vary considerably, the mean ratings suggest several trends. In general, participants gave higher importance ratings to features that allow users to do specific tasks such as file insertion and formatting. Lower ratings were given to features involving greater automation.

TABLE 2. *Mean ratings and standard deviations of importance levels of other features in word processors on a 0 to 8 point scale.*

Feature	Mean	SD
Inserting graphs, pictures, etc. from other files	6.34	1.8
Being able to copy formats from one section to another	6.04	1.9
Being able to apply many different types of tabs	6.01	1.8
Auto highlighting of improper grammar	5.88	2.1
Being able to apply different headers and footers within sections of the document	5.75	1.9
Auto spelling corrections	5.72	2.3
Being able to apply different numbers of columns in sections of a document	5.31	2.0
Auto capitalization of words	4.55	2.6
Automated outline creation	3.73	2.3
Having a “wizard” character pop-up to give direction (for example, Mr. Paperclip)	2.72	2.3

The final question asked participants to list aspects of word processors that have most decreased their productivity and need improvement. The most common response category (mentioned by 50.4% participants) concerned auto-formatting. Representative responses were: (1) “many auto features are more of a nuisance than a help,” (2) “does things for you without you wanting it at times,” (3) “auto-formatting takes over too much and is sometimes hard to overcome,” and (4) "there ought to be an obvious way to undo the automatic functions." Other frequent responses were: (1) spelling or grammar checker gives wrong suggestions, (2) setting of the margins and tabs is difficult, (3) more control wanted over auto-capitalization, and (4) annoyance with assistants/wizards.

DISCUSSION

While word processing applications have grown in complexity and sophistication since the introduction of microcomputers about 30 years ago, usability problems have not disappeared. One fundamental feature of contemporary word processors (and many other applications) is tool bars. Most toolbars contain only icons, and no text. The present results show a preference for icon-only toolbars over text-only toolbars (Garcia, Badre, and Stasko, 1994; Lodding, 1983). However, most preferred was having both icons and textual descriptors in toolbars. This result makes sense given that research suggests that people do not understand the meaning of some of the icons (even some commonly used ones). The added text definition or description provides an apparent way to learn what those less understandable icons mean.

Although some word processors provide text that relate to the icons, text descriptors are provided in awkward manner that is time consuming by requiring the user to hover the pointer over individual icons to deliver text in the form of bubble help or presented at another location on the screen. Some applications (but not many word processing applications) have toolbars with both icons and text (full words or abbreviations).

A disadvantage of including text in the

toolbars is that it would consume potentially valuable screen space. For example, it could limit the size of the document window and adversely affect word-processing performance. It could add visual clutter. However some of the problems might be overcome by allowing the user to easily switch off the text portion of the icon bar when not needed or after the icons have been learned. Of course, the text would need to be quite small when accompanying the icons. It should be displayed with a font that is maximally legible. Also, some of the text descriptors may need to be abbreviated due to space (letter count) restrictions. Research has shown that small text size and abbreviations can adversely affect usability (Sanders and McCormick, 1993). However, if needed, short abbreviations could be combined with a "hover over" maneuver to enlarge the space for a larger, longer description. Alternatively, users could be allowed to enlarge the size of the default text if it is too small for them. Despite the disadvantages of using small text and losing some screen space, the present study's data showed a preference for having the text visible and available as opposed to it being absent. Usability testing could help determine the most important parameters here and what tradeoffs and compensatory aspects should be incorporated.

The ratings of selected word processor features showed an interesting general pattern. The highest rated features suggested a desire to have some level of manual control in executing certain word processor tasks. Although all of the features participants rated involve some form of automation, the lowest rated items in the list (but not all of them) tended to be ones that involve more ongoing assistance. Also they tended to be features that make "assumptions" about what the user wants. Some of those assumptions may be incorrect, which could add to the higher workload and potentially add errors to the worked-on document. Together, this pattern of data suggests that people are having negative experiences in the ways the automatic features have been implemented.

The lowest ratings were for the wizard/assistant tool. While this tool has extensive and complex capabilities, the resulting actions may not be what the user wants or intends.

A somewhat similar account is given by the responses to an open-ended question asking to list the word processor features have caused them difficulty and need to be improved. Like the ratings, the open-ended responses tended to point to automatic features as being problematic. The responses suggested a dislike of automation because they often generated undesired responses that were difficult to change or turn off.

The intent of this article is not criticize particular word processing applications or the companies that manufacture and distribute the software, but rather to point out that current word processors still have substantial usability problems. Thus, despite many years and multiple versions and revisions of word processing software, the software has not approached a high level of maturity in terms of usability. Word processors have not evolved to the point where usability problems have been minimized.

This is unfortunate, because word processors are a main productivity tool for many occupations and production tasks. It is used in a wide range of tasks, as for example producing intellectual products. Poor usability can translate into wasted time, disturbance, and potentially the production of lower quality work documents. The goal of word processing software should be to enable writers to record their verbal ideas without being encumbered by the interface. Ideally the application should not get in the way of idea flow. In other words, the tool should be nearly "invisible" so as not to distract attention from the writing task.

REFERENCES

Garcia, M., Badre, A. N., and Stasko, J. (1994). Development and validation of icons varying gin their abstractness. *Interacting-with-Computers*, *6*, 191-211.

James, K., Lynk, D., Molinarie, and Caird, J.K., (1995). The comprehension and use of word processing icons. *Proceedings of the Human Factors and Ergonomics Society*, *39*, 941.

Leonard, S. D., Otani, H., and Wogalter, M. S. (1999). Comprehension and memory. In M. S. Wogalter, D. M. DeJoy, and K. R. Laughery (Eds.). *Warnings*

and Risk Communication. (pp. 149-188). London: Taylor and Francis.

Levine, J. D., and Wogalter, M. S. (August, 2000). On determining the relative importance of word processor functions and deficiencies. *Proceedings of the International Ergonomics Association & the Human Factors and Ergonomics Society Congress, 44* (1), 362.

Lodding, N. K. (1983). Iconic Interface. *IEEE Software*, March-April, 11-20.

Luximon, A., Ghambaryan, A., and Mehrabyan, A. (2003). Usability of operating Systems' (OS icons. *Proceedings of the International Ergonomics Association,* Seoul, South Korea.

Mehlenbacher, B., Duffy, T.M., and Palmer, J. (1989). Finding information on a menu: Linking menu organization to the user's goals. *Human-Computer Interaction, 4*, 231-251.

Mehlenbacher, Wogalter and Laughery (1999). On the reading of product owner's manuals: Perceptions and product complexity. *Proceedings of the Human Factors and Ergonomics Society*, *46*, 730-734.

Sanders, M. **S.,** & McCormick, E. J. (1993). *Human factors in engineering and design* **(**7 Ed.). New York: McGraw-Hill.

Schwalm, N. D., Shaviv, V., and Goldschmidt, G. (2000). Can icon animation enhance human performance. or is it just another gimmick? *Proceedings of the International Ergonomics Association & the Human Factors and Ergonomics Society Congress, 44* (1), 323-326.

Stainton-Skinn, J. (1982). Business and secretarial studies: The impact of word processors. *Journal of Vocational Education and Training*, *34*, 7-9.

Wiedenbeck, S., and Davis, S. (1997). The influence of interaction style and experience on user perceptions of software packages. *International Journal of Human-Computer Studies*, *46*, 563-588.

Wogalter, M. S., & Frei, R. L. (1990). Social influence and preference of direct-manipulation and keyboard-command computer interfaces. *Proceedings of the Human Factors Society*, *34*, 907-911.

Wogalter, M.S., Silver, N.C., Leonard, S.D. and Zaikina, H. (2006). Warning symbols (Chap. 12). In M.S. Wogalter (Ed.) *Handbook of Warnings* (pp. 159-176). Mahwah, NJ: Lawrence Erlbaum Associates.