Contextual online help: elicitation of human experts' strategies

Antonio Capobianco, Noëlle Carbonell

LORIA, Campus Scientifique, BP 239
F54506 Vandœuvre-lès-Nancy Cedex, France
tel.: +33 (0)3 83 59 20 32, fax: +33 (0)3 83 41 30 79
e-mail: {Antonio.Capobianco, Noelle.Carbonell}@loria.fr

Abstract: The paper reports the main results and conclusions of the analysis of eight dialogues between an expert and eight novice users of Word. The analysis of the expert's requests for contextual information together with her explicit references to, and implicit use of the various types of contexts indicate that the types of contextual information she exploits most are the progress of the current task execution, the software current state and the novice's current intention. Her help strategy, which differs greatly from standard didactic computer aided instruction approaches, encourages novices to adopt a "learning by doing" strategy through helping them to achieve the tasks which motivate their use of the software. This strategy relies mostly, for defining the informational content of help messages, on the short-term context and on a dynamic model of the novice's activities and goals, rather than on an individual (dynamic) or generic (static) cognitive user model.

1. Context, motivations and objectives

Online help to the use of standard application software intended for the general public is still highly unsatisfactory. Despite the continuous efforts of researchers and designers over the last twenty years, most novice users still prefer consulting experienced users to browsing available online help or paper user guides. In addition, specific training sessions and tutorials are still very popular among novices nowadays.

Since the beginning of the eighties, research has been developing along two parallel directions. One approach, which motivated many studies in the early eighties, when personal computers were becoming popular, is to design intuitive ("transparent") user interfaces, that is interfaces which aim at making interaction with any standard application software so easy as to eliminate the need for initial training to its use and operation (cf. the hopes underlying the design of direct manipulation, Shneiderman, 1983). The other main approach, which has been motivating continuous research since the mid eighties, is to provide users with contextual interactive online help as "intelligent" as human experts.

However, the behaviors of novice users, especially users in the general public, may explain the failures of both approaches. We briefly summarize the empirical findings which may explain these failures.

To be efficient, usable and actually used, online help has to overcome at least one major obstacle, that is the "motivational paradox" (Carroll, and Rosson, 1987) which greatly influences users' behaviors. Carroll and Rosson conclude from the analysis of various empirical data, that the general public, especially novice users, are reluctant both to explore a new software package and to learn how to use its functionalities efficiently. These users seem to be mainly concerned with achieving the tasks which motivate their interaction with the software; mastering its use does not rank among their motivations or goals. Therefore, they are not likely to take advantage of the exploration facilities offered by direct manipulation. More generally, they are liable to ignore the facilities for autonomous learning, especially learning by trials and errors, which intuitive user interfaces provide for familiarizing users with the use of a new software.

This paradox may also explain the failure of approaches which view online help as an interactive learning situation (e.g., Mallen, 1995). In particular, it may account for the tendency of the general public to ignore online tutorials (Carroll, and Mack, 1992), and their reluctance to consult online manuals, even in the form of databases (Fisher, Lemke, and Schwab, 1985; Rœstler, and McLellan, 1995), hypertexts (Cohill, and Williges, 1985; Moore, and Swartout, 1990), or hypermedia documents (Palmiter, and Elkerton, 1991; Harrison, 1995).

To take this paradox into account, design approaches have to consider online help as a specific human-computer interaction situation (Sellen, and Nicol, 1990; Duffy, Palmer, and Mehlenbacher, 1992). Contextual online help appears as an appropriate design framework in this context: providing the current user with the specific information (s)he needs to carry through the computer task in progress, and delivering it at the right moment, constitute a significant contribution to supporting users' interactive activities, thus meeting their expectations (Carenini, and Moore, 1993; Quast, 1993).

However, the adaptation of the content and presentation of help messages to individual needs and to their evolution in the course of interaction involves static and dynamic user models which are difficult to elaborate and to update from the interaction context. In particular, selecting the relevant meaningful contextual information and exploiting it appropriately, especially interpreting it accurately, are still crucial research issues, as illustrated by present implementations: products on the market are too crude to prove useful, and research prototypes too complex to be reliable, hence usable (cf., for instance, Paiva, Self, and Hartley, 1994; Brajnik, and Tasso, 1994).

On the one hand then, online help seems necessary for enabling users to learn how to use a new software efficiently. To convince oneself of this necessity, it suffices to consider the inability of intuitive user interfaces to achieve this goal, together with the fact that present standard software for the general public still includes online help facilities (cf., for instance, the Microsoft paperclip assistant) while online tutorials have completely disappeared.
On the other hand, this brief survey of research published on online help over the past twenty years suggests that users' behaviors constitute an insuperable obstacle to the design of online user support which assists users effectively in learning how to use a new software. The fact that since the mid nineties scientific publications on online help are scarce confirms this tentative conclusion, and implies that research efforts may have reached a dead end.

In order to contribute to overcoming this obstacle, we attempted to elicit the help strategies of human experts from the analysis of a corpus of expert-novice dialogues. This empirical study addresses two main issues:
- what type of contextual information do human experts actually use;
- and how do they exploit such information for providing novice users with efficient online help, that is help which users are prone to consult and which enables them to master the use of the given software?

Results of the analysis of written transcripts of the recorded dialogues are presented and discussed in paragraph 3. The experimental protocol and setup designed so as to obtain a corpus of meaningful verbal data are described in paragraph 2, together with the methods and procedures used for recording the experimental dialogues and interactions, transcribing them and analyzing them.

2. Collection, recording, transcription and analysis of meaningful empirical data

Experimental protocol and setup:
We analyzed eight expert-novice dialogues collected during an earlier study (Falzon, and Karsenty, 1997). Eight novice users performed twenty predefined formatting tasks on a given text, using Microsoft Word. They could communicate freely (over an intercom) with an expert user of Word who helped them to carry out the prescribed tasks. Expert and novice were in different rooms; in the first condition (condition A), the expert could view the novice's screen (via Timbuktu), while in the second one (condition B) she could not. The expert was asked to focus on helping novices to perform the prescribed tasks; she was requested to refrain from behaving as a tutor, that is from trying to "teach" them how to use Word efficiently.
Expert-novice dialogues were recorded; in addition, the novices' screen displays and hand gestures were videotaped. One week after the experimental session, they participated in a post-test which was intended for assessing their actual mastery of the use of Word.

Recordings and transcripts:
The textual corpus which was used for the analyses includes time-stamped transcripts of the dialogues together with descriptions of the novices' hand-gestures, their actions (on keyboard and mouse) and the effects of these actions on the user interface, the prescribed tasks which the novice's actions contributed to carrying out.

3. Analysis method and procedure

Our analyses rely on the characterization of the novice's and the expert's recorded speech acts according to two specific taxonomies.
The first one based on (Pilkington, 1992) categorizes the novice's requests. We shall not detail it here since the paper is focused on the analysis of the expert's speech acts exclusively.
We designed the second one with a view to characterizing the expert's utterances according to the type of context she used for selecting the information given to novices spontaneously or in answer to their questions:
- type of context used (current goal/intention and task/action of the novice, past interactions, dialogue exchanges, current state of the software, …); these categories were used for labeling both the explicit and implicit references of the expert to the interaction context; as for the type of contextual information used by the expert for selecting the information included in her help messages (cf., implicit references to the context), it was inferred from the contents of the messages themselves;
- type of information provided (not used here); category necessary for assessing the possible discrepancy between the explicit requests of novices and their interpretation by the expert;

– form of the utterance (answer, spontaneous utterance, question).

All utterances in the corpus were split up into their constituent speech acts; each speech act was then labeled using either taxonomy.

4. Results and interpretations

Analyses of the labeled corpus yielded qualitative and quantitative results which make it possible to gain an insight into the strategies implemented by the expert for filtering available contextual information and for exploiting the selected information.

We first present results obtained from the analysis of dialogues in condition B; these results concern the type of contextual information requested by the expert from the novice. As the expert had no visual access to the novice's environment, they provide meaningful cues on the type of contextual information involved in expert help strategies; they also give an idea of which type of context experts rate highest, need and use most.

Then, we summarize the findings stemming from the analysis of the expert's implicit and explicit references to the context in both conditions; these findings shed light on how the expert exploits the selected contextual information and, more generally, on the part such information plays in expert interactive help strategies.

4.1. Context filtering and selection

Requests for contextual information represent less than 2% of the expert's speech acts in condition A, and over 14% in condition B. In the latter condition, the number of such requests is eight times as great as in condition A (81 versus 10). These results denote the high implication of contextual information in the expert's help strategy, and then, the importance and necessity of such information for achieving "intelligent" help to the use of a new software. The crucial part played by contextual knowledge in the expert's help strategies is confirmed by the fact that she feels the need to request such information even in condition A, although she has access to the novices' screen displays.

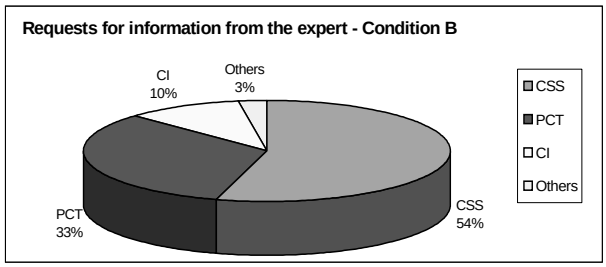


Figure 1: The expert's requests for information. Condition B.

CSS: current state of the software as displayed;
PCT: progress of the current task execution;
CI: current intention of the novice.

As regards the type of contextual information sought by the expert, figure 1 shows that most requests in condition B (i.e., 87%) represent attempts to elicit from the novice either the current state of the software application as displayed on the screen (CSS, 54%) or the progress of the current task execution (PCT, 33%). 10% of the requests only are queries about the novice's current intention (CI), that is the intention that motivates his/her last (or current or next) sequence of related actions; by "sequence of related actions" we mean the sequence of actions required for carrying out either a simple task or a single step in the execution of a complex task which the novice intends to carry out or is currently trying to achieve.

On the other hand, all requests for contextual information formulated in condition A aim at eliciting the novice's current intention, although the expert had a priori knowledge of the list of formatting tasks the subjects had to perform. Let us note that this finding illustrates the difficulty (or perhaps even the impossibility) of identifying users' intentions from the sole analysis and interpretation of the interaction trace; this difficulty is even greater in the case of novice users who do not master the use of the software and make frequent usage errors.

These results lead to the conclusion that the interaction context represents a source of dynamic knowledge which is essential for the implementation of expert interactive help. They also suggest that the strategy adopted by the expert for familiarizing novice users with the use of a software involves helping them to achieve the tasks which motivate their use of the software; most of the expert's requests for contextual information in condition B concern the current state of the software and the progress of the current task execution; this dynamic knowledge is essential for guiding novice users through the execution of the computer tasks they intend to achieve.

Therefore, one can tentatively assume that the overall strategy of the expert is to encourage and help novices to familiarize themselves with the use and operation of the software by adopting the autonomous learning strategy called "learning by doing" which implies "learning by trials and errors". Help systems implementing such a strategy are likely to be effectively used by novices and well accepted by them (subjective satisfaction), in-as-much as they implicitly take account of the motivational paradox.

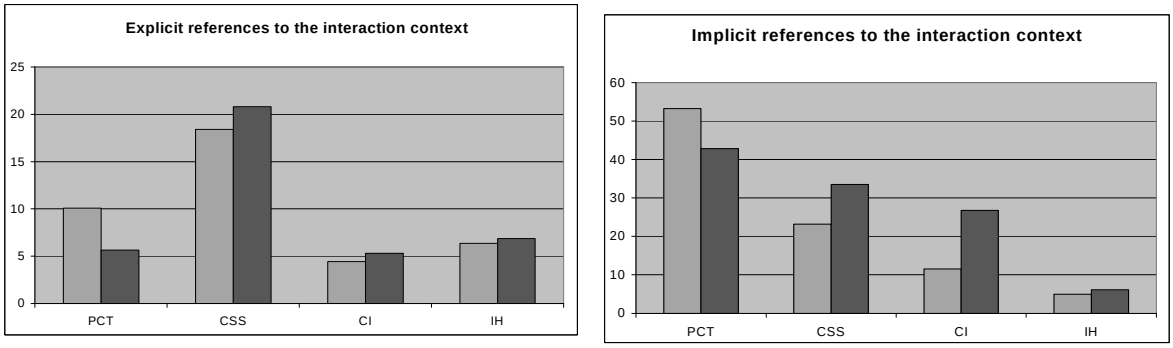
The analysis of the expert's use of contextual information, which is reported in the next paragraph (cf. 3.2), contributes to validate this assumption; it also provides supplementary information on the expert's help strategy.

4.2. Exploitation of contextual information

The results discussed in this paragraph stem from the analysis of both the explicit references of the expert to the interaction context and her implicit use of contextual information, that is her taking such dynamic knowledge into account while designing the information content of the help messages she formulates either spontaneously or in answer to the novice's questions.

We were expecting to observe numerous occurrences of explicit references to the interaction context. References to this dynamic knowledge shared by the novice and the expert may indeed prove useful either for clarifying and illustrating help information (especially procedural information), or for relating this information to the user's current activity. However, more than 40% of the expert's speech acts do not include any explicit reference to the interaction context (46% in condition A and 41% in condition B).

The only finding which is in keeping with our expectations is the fact that the current state of the software (CSS), the progress of the current task execution (PCT) and, to a lesser extent, the novice's current intention (CI) are types of contexts frequently referred to by the expert (cf., figure 2). The expert uses this information mostly to relate the procedural information she gives to the novice both to the current task execution and to the latter's current intention; thus, she helps novices to "learn" the representation of the task domain implemented by the designer (i.e., the semantics of the software functions) or, at least, to adjust their a priori mental representations of the task domain and the software capabilities to the designer's one.



Figures 2 and 3: Context exploitation. Percentages of explicit references to (left), and implicit use of (right), contextual information, computed over the total number of the expert's speech acts.

Condition A Condition B

(PCT: progress of task execution; CSS: software state; CI: novice's intention; IH: interaction history)

Figure 2 also shows that the expert seldom refers to the previous actions of the novice (cf., the interaction history noted IH); this suggests her reluctance to use the dynamic knowledge provided by an individual cognitive model of the current user; such models are currently implemented in sophisticated didactic software or adaptable and adaptive user interfaces. Her prevailing strategy exploits a dynamic model of the novice goals and activities rather than an individual (dynamic) or generic (static) cognitive user model. She resorts to a generic task-oriented (static) cognitive user model only for preventing or correcting the novices' usage errors.

On the other hand, the implicit use of contextual information conforms to our expectations, since over 90% of the expert's speech acts include implicit references to the interaction context (95% in condition A and 93% in condition B). On the whole, the expert uses contextual information extensively (either explicitly or implicitly); however, she prefers using it implicitly to referring to it explicitly. Generally speaking, taking account of the interaction context in the definition of the information content of help messages appears as a key feature of the help strategies of human experts; this feature should then be implemented in online help systems which aim at emulating human expertise; on the other hand, designers need not bother to insert explicit references to the context in help messages.

The various types of contexts which were inferred from the help information included in the expert's speech acts are presented in figure 3. These results, which are roughly similar to those described in figure 2, bring out the main characteristics of the expert's strategy which can be summarized as follows. The design of the information content of the majority of help messages is mainly influenced by the progress of the current task execution, PCT (over 40% of the expert's speech acts in either condition). It is also influenced, but to a lesser degree, by the current state of the software application (CSS) and the novice's current intention (CI). As the knowledge and use of PCT imply those of the novice's current intention (CI) and the current state of the software application (CSS), the influence of PCT on

the content of help messages also implies those of CI and CSS. These data confirm and consolidate the conclusions of the analysis of the types of contextual information involved in explicit references to the context, all the more so as the difference between the implicit use of the interaction history (IH) and the use of other contexts is much greater than the corresponding difference in figure 2 (regarding explicit references to the context).

The results described in this paragraph confirm the tentative conclusion presented in paragraph 4.1, namely that the expert's strategy for helping novices to learn how to use a new software involves assisting them in carrying out the computer tasks they have a mind to achieve.

5. Conclusion

The paper reports the main results and conclusions of the analysis of eight dialogues between an expert and eight novice users of Word. The analysis of the expert's requests for contextual information together with her explicit references to, and implicit use of the various types of contexts indicate that the types of contextual information she exploits most are the progress of the current task execution, the software current state and the novice's current intention. Her help strategy, which differs greatly from standard didactic computer aided instruction approaches, encourages novices to adopt a "learning by doing" strategy through helping them to achieve the task which motivate their use of the software. This strategy relies mostly, for defining the informational content of help messages, on the short-term context and on a dynamic model of the novice's activities and goals rather than on an individual (dynamic) or generic (static) cognitive user model.

Such a strategy should win user acceptance as it takes the "motivational paradox" into account. The results of the post-test that the subjects performed a week after the experiment demonstrate its efficiency as regards the learning of the software usage. However, its efficiency/utility should be assessed more thoroughly by comparing it to standard interactive tutorial approaches and to non contextual help strategies. We are currently implementing it in a mock-up which will be used to assess its efficiency empirically.

6. References

Brajnik, G., and Tasso, C. (1994). A flexible tool for developing user modelling applications with non monotonic reasoning capabilities. *International Journal of Human-Computer Studies*, 40(1), pp. 31-62.

Carenini, G., and Moore, J.D. (1993). Generating explanations in context. *Proceedings of the ACM Conference on Intelligent User Interfaces*, Orlando (FL), New York: ACM Press, pp. 175-182.

Carroll, J.M., and Mack, R.L. (1992). Learning to use a word processor: by doing, by thinking, and by knowing. In J. Thomas, and M. Schneider (Eds.), *Human Factors in Computer Systems*, Norwood, NJ: Ablex Publishing Corporation, pp. 13-51.

Carroll, J.-M., and Rosson, M.B. (1987). Paradox of the active user. In J.M. Carroll (Ed.), *Interfacing thought: cognitive aspects of human-computer interaction*, Cambridge (MAS): MIT Press, pp. 81-111.

Cohill, A.M., and Williges, R.C. (1985). Retrieval of help information for novice users of interactive computer Systems. *Human Factors – Journal of the Human Factors and Ergonomics Society*, 27(3), pp. 335-343.

Duffy, T.M., Palmer, J.E., and Mehlenbacher, B. (1992). *Online help: design and evaluation*. Norwood, NJ: Ablex Publishing Corporation.

Falzon, P., and Karsenty, L., Eds. (1997). *Dialogue et coopération*. Rapport final de contrat, PRC Sciences Cognitives, Paris : CNAM.

Fisher, G., Lemke, A., and Schwab, T. (1985). Knowledge-based help systems. *Human factors in computing systems: Proceedings of the CHI'85 Conference*, New York: ACM Press and Addison Wesley, pp. 161-167.

Harrison, S. (1995). A comparison of still, animated or non illustrated on-line help with written or spoken instructions in a graphical user interface. *Human factors in computing systems: Proceedings of the CHI'95 Conference*, New York: ACM Press and Addison Wesley, pp. 82-89.

Mallen, L.C. (1995). Designing Intelligent Help within Information Processing Systems. PhD Thesis, Leeds University, UK.

Moore, J.D., and Swartout, W.R. (1990). Pointing: a way toward explanation dialogue. *Proceedings of the eighth national conference on artificial intelligence*, Boston, AAAI Press: Menlo Park, vol. 1, pp. 457-464.

Paiva, A., and Self, J. (1994). A learner model reason maintenance system. In Cohn A.G. (Ed.), *ECAI-94: European conference on artificial intelligence*, Amsterdam: J. Wiley and Sons, pp. 193-196.

Palmiter, S., and Elkerton, J. (1991). An evaluation of animated demonstrations for learning computer-based tasks. *Human factors in computing systems: Proceedings of the CHI'91 Conference*, New York: ACM Press and Addison Wesley, pp. 257-263.

Pilkington, R.M. (1992). Question-answering for intelligent on-line help: the process of intelligent responding. *Cognitive Science*, 16, (4), pp. 455-491.

Quast, K.J. (1993). Plan recognition for context sensitive help, *Proceedings of the ACM Conference on Intelligent User Interfaces*, Orlando (FL), New York: ACM Press, pp. 89-96.

Roestler, A.W., and McLellan, S.G. (1995). What help do users need? Taxonomies for on-line information needs and access methods. *Human factors in computing systems: Proceedings of the CHI'95 Conference*, New York: ACM Press and Addison Wesley, pp. 437-441.

Sellen, A., and Nicol, A. (1990). Building user-centered on-line help. In B. Laurel (Ed.), *The art of human-computer interface design*, New York: Addison Wesley, pp. 143-153.