(4)

N00014-88-J-1072

Learning from Error

Colleen M. Seifert
UCSD and NPRDC
Edwin L. Hutchins
UCSD

Introduction

Most studies of error focus on its reduction or elimination. Clearly, there are many steps that can be taken to avoid or prevent the occurrence of errors. Yet in human systems, error is inevitable. This is commonly argued on the grounds that people can become tired, or confused or distracted, or fail to attend to their work, or are in some other way inherently fallible. All of these are real factors in many errors, of course, but for systems of cooperative work in the real world, there may be a less disparaging and more fundamental reason for the inevitability of error. This is that such systems always rely on learning on the job, and where there is the need for learning, there is potential for error.

A naturally situated system of cooperative work must both produce the intended result of its process and reproduce *itself* at the same time. Such cooperative systems may change over time, be reorganized, change the things they do, and change the technology they utilize to do the job. Even if tasks and tools could be somehow frozen, changes in personnel are certain over time. Most commonly, relatively expert personnel are gradually lost while relatively inexpert personnel are added. Even if the skills required to do the job can be taught in schools, the interactions that are characteristic of cooperative work can generally only be learned on the job.

Designing for Error

Norman (1983, 1986, 1987) argues that because error is inevitable, it is important to "design for error." Speaking of designers, Norman says, "Inadvertently, they can make is easy to err and difficult or impossible to discover error or to recover from it. " (1987, Ch.5:24). Norman suggests that designers of artifacts should design to minimize the causes of error, make it possible to "undo" errorful actions, and make it easier to discover and correct errors. These same goals are appropriate for designers of cooperative work systems, but here we can go further. Each of Norman's suggestions is aimed at protecting the current task performance, yet in the broader perspective of production and reproduction in cooperative work, it would be nice if the response to error in the current task could also in some way *benefit* future task performance. That is, another aspect of designing for error might be designing systems that can more easily learn from their errors.

That would give us two major classes of design goals with respect to errors. First, design to eliminate, avoid, or prevent errors wherever possible. Second, design to take full advantage of any errors that do occur. The goal is to facilitate learning from errors so that future errors become less likely. As career trajectories take experienced members out of the work group and expertise is lost from the system, the likelihood of error may increase. The potential advantage of designing for error is that this increase in likelihood of error may be offset by the decrease in likelihood of error due to learning by the remaining and new members of the cooperative work group. The prevention of error is clearly important and has received a great deal of attention in the past. In this paper, we will examine the less often considered aspects

A-1

of the organization of cooperative work settings that can become important once an error has occurred.

The Navigation of Large Ships.

The response of systems of cooperative work to error came to our attention in our studies of navigation teams aboard large ships (Hutchins, in press; Hutchins, n.d.a). At all times while a naval vessel is underway, a plot of its past and projected movements is maintained. The information gathered and processed by the navigation team supports the decisions of the conning officer who is responsible for the ship's movement. Day and night, whenever a ship is neither tied to a pier nor at anchor, navigation computations are performed. Most of the time the work of navigation is performed by one person working alone, however, when a ship leaves or enters port or operates in any other environment where maneuverability is restricted, the computational requirements of the task may exceed the capabilities of any individual. In such circumstances, the navigation duties are carried out by a team of individuals working together.

In addition to satisfying the immediate navigation needs of the ship, navigation teams have developed under the constraints of maintaining a working system in a state of readiness, allowing frequent replacement of individual team members and providing a task performance environment in which the job can be learned by doing it. These characteristics are shared by many real world settings of cooperative work.

We observed the activities of navigation teams aboard several ships operating in both solo and group performance configurations and made detailed recordings of the behavior of individual team members[1]. In spite of the fact that all of the navigation

---

[1] The two authors have closely observed the performance of navigation teams engaged in actual (not simulated) operations aboard several ships. In all cases, notes were made during the course of observations. Extensive audio recordings of the team in action were made aboard two ships and audio and

teams we observed functioned satisfactorily, close examination of their operation revealed surprisingly high rates of error. However, observing only the final output of the teams, one would not suspect that many errors were being made. In fact, while many errors were committed, virtually all of them were detected and corrected within the navigation team.

Facilitating Learning from Error

In order to benefit from errors that do occur, these errors must de detected, diagnosed as to their cause, and corrected with useful feedback. The next sections examine these three processes as they are facilitated by particular system characteristics.

Detecting error

Error detection may require considerable resources. Our observations about the conditions under which errors are detected indicate that the following elements are necessary for error detection and may play a role in diagnosis and correction as well.

- Access: In order to detect an error, the detector must have access to the errorful behavior or some indication of it.
- Knowledge/expectation: The detector must have knowledge of the process being performed or some expectation about its correct outcome with respect to which the observed process or outcome can be judged discrepant.
- Attention: The detecting entity must attend to the errorful behavior and monitor it in terms of expectation.
- Perspective: Different perspectives can result in focus of attention on different aspects of the task, consequently affecting the nature of the discrepancies in performance that are noticed.

---

video recordings were made aboard another ship. In addition, one of the authors, Hutchins, has many years of practical experience in ocean navigation.

In the world of navigation, as in many other systems, novices begin by doing the simplest parts of the collaborative work task. As they become more skilled they move on to more complex duties, making way for less skilled people behind them. This movement defines a career trajectory for individuals through the roles of the work group. An interesting aspect of the navigation setting is that the career trajectory for individuals follows the path of information through the system in the team's most basic computation, *position fixing*. The simplest jobs involve gathering sensed data, and the more complex jobs involve processing that data. As a consequence of this alignment of career trajectory with the path of information through the system, if one has access to an error, one also has knowledge of the processes that may have generated it because one has already, at an earlier career stage, performed all those operations. The overlap of access and knowledge that results from the alignment of career path and data path is not a necessary feature of these systems, nor is it apparently an intentional one here. It does, however, give rise to especially favorable conditions for the detection and diagnosis of error.

The attention required to detect error may be facilitated or even required by the nature of coordinated tasks. Many errors in earlier processing are detected by the plotter when visual bearings are plotted. In part this is because the plotting procedure itself is designed to detect error. Any two lines of position define the location of the ship, but a position "fix" always consists of three lines of position, whose intersection forms a small triangle. If any of the three is in error, the triangle will become larger. Thus the nature of the plotter's task itself makes errors in the bearings evident. But there is something more as well. The way the plotter thinks about the bearings and uses them in his task is different than the way the bearing observers think about the bearings. For the bearing observer, the bearing may be no more than a string of three digits read from a scale in a telescopic sight. It is not necessary for the bearing observer to think of the directional meaning of the number. In contrast, the plotter's job is to recover the directional meaning of the reported bearing and combine the meanings of three bearings to fix the position of the ship. Different jobs in the team require attention to different aspects of the computational objects, so different kinds of error are likely to be detected (or not) by different members of the team.

Many errors are detected by team members who are simply monitoring the actions of those around them. Not only is each member of the team responsible for his own job, each seems also to take responsibility for all parts of the process to which he can contribute. Since detection depends upon access, however, the more the activities of the team members are conducted where they can be observed (or overheard) by others the higher the potential rate of error detection. Detection also requires attention which may be a scarce resource. One of the consequences of high workloads may be both an increase in the rate of error itself due to the effects of stress, and a reduction in the rate of error detection due to the reduction of resources available for monitoring the actions of others.

On some ships the job of making a global assessment of the quality of the work of the navigation team is institutionalized in the role of the evaluator. This evaluator is a qualified navigation practitioner who is not engaged in doing the navigation computations themselves, but instead monitors the process by which the computation is performed and the quality of the product. There is a tradeoff here between using the evaluator's processing power to do the computations themselves thereby possibly lowering error rates while risking lower error detection rates, versus keeping the evaluator out of the computations thereby possibly increasing errors rates while detecting more of the errors that are committed.

The important structural property of the evaluator role is that the evaluator has access to and knowledge of the performance of the task, but does not participate in its performance. Instead, the evaluator *attends* to the way the task is done and specifically monitors the performance for error. The evaluator builds into the system some attention to how well the result of the computation fits the physical world it measures, an aspect of the system's behavior that would not otherwise be reliably present. This same strategy is recognizable in the mandated role of the captain of the ship, or that of the senior captain in a commercial airline cockpit. These people are task monitors rather than task doers (Miyake, 1982).

Diagnosing errors

Not all recoveries from error are instructional in intent or consequence. Because some recovery methods are used simply to complete the task, there may be no need to diagnose the cause of the error in order to discover how to recover from it. However, other error recovery strategies involve the *diagnosis* of the source of the error, and perhaps explicit demonstration of the correct solution. Diagnosing error depends on an understanding how the error may have been generated. This may require modelling of the reasoning processes of the person who committed the error. The *distribution of knowledge* characteristic of the navigation system in which access to error and the knowledge of its causes are aligned insures that most errors that are detected will be detected by people who already have experience with the operations that led to the error. Familiarity with the task assists in modelling other's understanding to determine where the generation of the error may have occurred. Errors indicate in very specific ways exactly what information or ability is missing in the current knowledge state of the novice. Consider this example:

A novice navigator was asked, "How far west shall we go to get back to harbor at 1600?" The ship was directly west of the harbor entrance (time = 1200). He paused, then measured the distance the ship could go in an hour and marked it with a compass; he then marked four hour-lengths from the harbor entrance on the chart. This position lay far west of the ship's current position. A more experienced *quartermaster was observing, and said, "If he wants to be back by 1600, he's not going to go west from now until he hits this point!"*

In the task of diagnosis, the expert attempts to determine what may be the cause of the novice's error; to do this, he must utilize meta-knowledge about what the task requires and what the novice is likely to know. A solution procedure would be to measure the current distance to land from the ship, subtract the time to return from this location from the four hours, and then split the remainder to continue west for half of it. The novice appears to know part of the solution procedure: to mark the distance travelled with a distance preserving tool (the divider) and compare the distance to a scale on the side of the chart which provides a transformation into miles. However, this procedure is incorrectly applied to the problem when he focuses on the ability of the ship to travel a particular distance within the time interval rather than on determining the particular distance already traveled from the end point. The response of the observer indicates he is modelling the reasoning process of the novice, as he points out that the solution being generated by the novice is not going to solve the stated problem of returning by 1600.

The particular solution generated by the novice also provides information about what knowledge he may be lacking or what elements are problematic for him; the expert can utilize the error to gear the explanation to the novice's current knowledge. By modelling the novice's understanding of the task, the expert may be able to determine where the novice went wrong in his reasoning and how to describe the solution in a way that is useful to the learner. For example, in the above problem, the expert may recognize that the novice started out by solving a more familiar problem; namely, how far are we from location X? The solution to that problem is related, but the novice did not know how to pose the course recommendation in a form that connected to this solution. By understanding the way in which the novice was attempting to solve the problem, the next step of correction is assisted because the expert can gear the presentation of a better solution in a way that is more understandable and memorable to the novice.

Correcting errors

Beyond the purpose of correcting an error that has occurred, a consequence of having engaged in the activities of detecting and/or diagnosing the cause of an error may be that the person doing the detecting comes to a new insight about the operation of the system. This is true whether the error was committed by oneself or someone else, and it may be particularly important for novices who detect other's errors. Further, every instance of correction is practice in the skills of detection and confirmation of knowledge which may save the system from consequences of future error.

Feedback on how to correct the error is extremely important to the learning process. Without correction, further performance acts to increase familiarity with the error path, thereby increasing the tendency towards error (Anderson et.al, 1984). However, if competing solutions are presented as feedback or can be inferred by the learner from the feedback, the error serves to direct the learning focus towards information that has been demonstrated to be missing from the novice's knowledge base. Even when the feedback lacks instructional content, it could contribute to the refinement of understanding task requirements that may not be apparent from correct performance alone. Such corrections help the learner to induce the principles that define correct performance.

This can be especially important with concepts that must be inferred from cases rather than explicitly stated. Because relevant information for a decision may not be explicitly observable or explicable by an expert, novices have to infer the domain information from experience in a variety of situations, guided by error correction on specific failures. Where there is a solution space to be explored, response to error can guide the discovery of the concept underlying the solution. The result may be a directed search through the information about a task guided by the particular errors made by the novice during performance. Thus, the implicit nature of domain knowledge in the navigation task motivates learning through error. Novices are allowed to do their best, and are provided directed instruction on the particular errors they make.

However, feedback with little instructional content may not be as helpful as a more complete demonstration or instruction. In navigation teams, error feedback is sometimes reduced to a contentless complaint or an exhortation to do better. Such limited feedback may be of little use to the person who has committed the error. However, it may be the only response the error detector can provide. Because the detector is also involved in a subtask, he may not have the time, processing resources, or communication channels required for the composition and delivery of appropriate instruction. Further, because of the ongoing nature of the task, these resources must be available to the person providing correction at or near the time that the error is committed in order to fully benefit from the specifics of the error commission setting. This represents a tradeoff between apprentice training systems, where two people perform the task of one, and redundancy within cooperative systems, where each has a separate task to complete but shared knowledge allows some correction of errors.

Learning from one's own mistakes is an obvious case of improvement of future performance from correction. Particularly in the early stages of acquisition, the correction of errors may play a significant role in improving performance. In addition, meta-knowledge about errors may be transferred through the cooperative process. Contentful corrections may help the novice learn recovery strategies that can be applied to self-detected errors. And, through the correction of errors, the novice may internalize the processes of error detection, diagnosis, and correction.

Additional learning opportunities are provided by the errors others make. When an error is detected and corrected in the context of collaborative work, many participants may witness and benefit from the response to error. Depending, again, on the *horizon of observation* (who has access to what behavior of others), error and correction may provide a learning context for many of the participants. The novice has the opportunity to observe others' errors, witness their correction, and participate in detecting the errors of others. Thus the socially distributed task gives a participatory role to the novice in all areas of task performance that he is physically able to perceive. Thus, the value of a response to error for future performances may depend upon the horizon *of observations for various members* of the team. Witnessing such a correction may be of value as well to those who are already competent in the task in which the error occurs if they will subsequently be in a position to detect and correct such errors. They can learn about how to provide useful feedback by watching the corrections of others, leading to an improvement in subsequent learning for others in the system.

Through errors, novices learn by being corrected and instructed by more advanced members of the team, by self-detecting, diagnosing, and correcting, and by observing the errors and corrections of others in close proximity, by participating in the detection, diagnosis and correction of others. The avenues of knowledge acquisition investigated by novice navigators are much richer as a result. By participating in the process of error-based learning as a performer, an observer, and a teacher, the novice increases the number of learning experiences available for acquisition and deepens understanding of the lesson through interaction in several participatory roles. In addition, other learners provide models of the learning process itself, and this meta-knowledge may be helpful to novices in forming expectations about their own performance.

Discussion

Tradeoffs are prevalent throughout this analysis: designing systems so as to benefit from error that does occur requires maximizing the ability to learn from the errors. However, improving the detection, diagnosis, and correction capabilities within a system often has other consequences for system performance. For example, *perspective* can be affected within a system to improve error detection. The bearing observers frequently take the perspective of "meter readers" in their performance, ignoring the physical world perspective which would give them some top-down expectations regarding the plausibility of the readings. Consequently, they often fail to detect errors that they *could* recognize using a physical direction perspective. Bearing observing *requires* no thought about how the information is used -- the numbers are simply reported. Despite the fact that they *know* the intended use of the number, they have little motivation to think about the number in terms of its meaning in the coordinate space of the chart. Consequently, error is propagated through the system past the point where it could logically be detected and corrected.

However, the task system could be redesigned to encourage the directional perspective in the bearing observers. One method which would improve self-detection of these errors is to remind the observers of how the information will be used by placing an artifact indicating the directional coordinate space into their work environment -- for example, have an indication of the full 360 degree representation available on the gyrocompass instead of the partial display currently provided. Such a cognitive artifact would serve to *remind* the takers of the coordinate space and therefore of the plausibility of the reported number in that space. To the extent that the bearing observer adopts the perspective of "thinking direction", he will be better able to utilize plausibility information to detect his own or other's errors. But there is a tradeoff involved in affecting perspective: knowing the intended use and plausibility of the reading may influence the perception of the reading. The point is that in this as in many other design decisions, the choice of whether to cut error rates or diminish the separation of perspectives is simply a tradeoff to be decided based upon environmental features or system goals.

The analysis provided other observations of design tradeoffs in cooperative systems; for example, the evaluator observed on one ship served to detect errors but did not participate in the computation. The cost to this separation of evaluation from computation is first that error is often detected much later in the computational process than need be if evaluation were performed as the processing occurred; and secondly, that the computational advantage of including this potential participant is lost. Other tradeoffs include: the horizon of observation, which allows error detection but increases distraction; the distribution of knowledge, which improves diagnosis but increases costs due to redundancy of training; and error rates, which allow opportunities for learning but cost in current task performance, and in resources for detection and recovery. Under some conditions, these costs can be offset to some extent by the benefits derived from the process of learning from errors through detection, diagnosis, and correction of errors. Achieving these benefits is not an automatic result of error, but requires ways to organize systems that are more likely then others to notice, recover from, and change future performance based on errors. At present, we know of no way to quantify the tradeoffs involved; however, recognizing their nature and identifying the properties of cooperative systems that affect them seems like a useful first step.

In conclusion, errors will occur in any system of human behavior; however, cooperative systems can benefit from the unavoidable occurrence of error if they are designed to facilitate learning from error. The intent of this research was to examine how learning from errors takes place in a natural setting, and in particular, how the cooperative task setting fosters learning within a complicated computational task. The results of the analysis point to design features of the navigational environment that are well-suited to learning from error. The demands on system organization include not only that the task be completed without major error, but that the system replicate itself and train novice team members while participating in the task. The navigation system appears well-suited to allow novices to perform on the job with little prior training, to allow errors to indicate where instruction is necessary, and to allow detection, diagnosis, and correction of errors while avoiding their propagation into the final decision phase of the task. An analysis of the task properties has identified particular features of this successful task environment, and the criteria identified can be used to design, *to analyze, and to intervene in problem situations within cooperative task* systems.

References

Anderson, J.R., C.F. Boyd, R. Farrell, & B.J. Reiser. (1984). Cognitive Principles in the design of computer tutors. Office of Naval Research Technical Report #84-1.

Hutchins, E. (in press) Learning to navigate in context. In J. Lave and S. Chaiklin (Eds.) *Context, cognition, and activity*. Cambridge: Cambridge University Press.

Hutchins, E. (n.d.a) Technology in the cooperative work of navigation. Paper prepared for a conference on Technology and cooperative work, Tucson Arizona.

Miyake, N. (1982) .Constructive interaction. CHIP technical report #113, Center for human information processing, University of California, San Diego, La Jolla, CA

Norman, D.A. (1983). Design rules based on analyses of human error. *Communications of the ACM,4*, 254-258.

Norman, D.A. (1986). Cognitive engineering. In D.A. Norman and S.W. Draper (Eds) *User centered system design: New perspectives on human-computer interaction*. Hillsdale, NJ: Erlbaum Associates.

Norman, D.A. (1987). *The psychology of everyday things*. New York: Basic Books.