Applied Ergonomics 53 (2016) 258–266

Contents lists available at ScienceDirect

Applied Ergonomics

journal homepage: www.elsevier.com/locate/apergo

Effects of monetary reward and punishment on information checking behaviour

Simon Y.W. Li [a, *], Anna L. Cox [b], Calvin Or [c], Ann Blandford [b]

[a] Department of Applied Psychology, Lingnan University, WYL Building, 8 Castel Peak Road, Tuen Mun, Hong Kong SAR
[b] UCLIC, University College London, 2nd Floor, 66 – 72 Gower Street, London, WC1E 6EA, UK
[c] Department of Industrial & Manufacturing Systems Engineering, University of Hong Kong, Haking Wong Building, Pokfulam Rd., Hong Kong SAR

ARTICLE INFO

Article history:
Received 7 July 2015
Received in revised form
18 October 2015
Accepted 21 October 2015
Available online 6 November 2015

Keywords:
Error
Reward
Punishment

ABSTRACT

Two experiments were conducted to examine whether checking one's own work can be motivated by monetary reward and punishment. Participants were randomly assigned to one of three conditions: a flat-rate payment for completing the task (Control); payment increased for error-free performance (Reward); payment decreased for error performance (Punishment). Experiment 1 ($N = 90$) was conducted with liberal arts students, using a general data-entry task. Experiment 2 ($N = 90$) replicated Experiment 1 with clinical students and a safety-critical 'cover story' for the task. In both studies, Reward and Punishment resulted in significantly fewer errors, more frequent and longer checking, than Control. No such differences were obtained between the Reward and Punishment conditions. It is concluded that error consequences in terms of monetary reward and punishment can result in more accurate task performance and more rigorous checking behaviour than errors without consequences. However, whether punishment is more effective than reward, or vice versa, remains inconclusive.

1. Introduction

Instead of focusing on how to prevent errors, error management considers how errors can be detected and recovered (Zapf and Reason, 1994). Error detection becomes important in mission-critical situations; for example, the NASA Mars Climate Orbiter, which costs about US$190 million, approached Mars at an erroneously low altitude and disintegrated. A contributing factor was that the wrong navigation information was sent due to an undetected incorrect unit conversion. Error detection is essential in safety-critical procedures such as medication administration: e.g., can a nurse easily detect any errors when programming an infusion pump? Can a doctor readily detect errors when prescribing medications using a computerized provider order entry (CPOE) system?

Despite the importance of error detection, there are only a handful of studies devoted to the topic: they range from experiments using statistical problem-solving tasks (Allwood, 1984) to computer application usage (Rizzo et al., 1987); and from diary studies (Sellen, 1994) to quasi-experimental field studies (Zapf et al., 1994). These studies are primarily concerned with whether errors are detected and whether certain types of error (e.g. slips or mistakes) are detected more easily than others. In this paper, we add to the understanding of error detection by treating checking behaviour as an essential part of the detection process and ask the questions: can people be encouraged to check their own performance when it is discretionary? Can checking behaviour be motivated by reward and punishment?

A number of theoretical models of error detection have been proposed offering ways to describe detection processes. Reason (1990) proposes three main ways in which error detection can occur: (1) self-monitoring – detection happens through monitoring one's own performance; (2) environmental cueing – errors detected via cues/feedback in the environment; and (3) detection by others – errors detected by other people. Based on action control theories (e.g. Norman, 1981), Sellen (1994) proposes a similar taxonomy of error detection processes that includes action-based detection, outcome-based detection, detection by limiting function, and detection by other people. Action-based detection involves discovering errors by erroneous actions, outcome-based detection relies on the consequences of actions, and detection by limiting function is achieved via feedback from external constraints

* Corresponding author. WYL-306, WYL Building, Lingnan University, 8 Castle Peak Road, Tuen Mun, Hong Kong SAR.
E-mail addresses: simonli2@ln.edu.hk (S.Y.W. Li), anna.cox@ucl.ac.uk (A.L. Cox), klor@hku.hk (C. Or), a.blandford@ucl.ac.uk (A. Blandford).

http://dx.doi.org/10.1016/j.apergo.2015.10.012
0003-6870/

in the environment. More recently, Blavier et al. (2005) developed a model of error detection based on prospective memory (Ellis, 1996) and highlights the importance of intention formation and retention in detecting errors.

A common theme that emerges from the above theoretical models is that regular monitoring or checking of one's own performance forms an essential component in error detection processes. This is supported by empirical studies, for example, Allwood (1984) found that when his participants were asked to solve statistical problems, standard check, which involves checking of a solution for scrutiny purposes rather than correctness, was one of the main error detection processes. Nyssen and Blavier (2006) studied error detection behaviour in anaesthetists and obtained similar findings suggesting that anaesthetists detect their own errors largely by routine monitoring of the environment (standard check). Furthermore, standard check was adopted more among young anaesthetists whereas a wider range of detection processes was adopted by experienced anaesthetists. Taking together the findings on standard check, they suggest that it is a frequently adopted detection process with reasonable effectiveness; however, it seems like a basic method and more sophisticated processes are employed by experts with more domain experience.

Research in human-computer interaction (HCI) has shown that people are sensitive to information access cost, i.e. how easy or difficult to see or obtain a piece of information, when checking information. In a simulated video programming task, it was found that when information about the to-be-recorded programme was made slightly difficult to access (in this case, the participants had to click on a grey box to reveal the information), the participants were less likely to check for the programme information that was even only one mouse-click away (Gray and Fu, 2004). More recent HCI studies have examined the effectiveness of different checking methods on catching data-entry errors (Barchard and Pace, 2011; Barchard and Verenikina, 2013). Double entry (data were re-entered a second time), read aloud (data entries were checked while they were read aloud by another person) and visual checking (verify by sight if there were any mismatched entries) were compared and both studies found that double entry was the most effective in catching data-entry errors. Moreover, the superior effectiveness of double entry was found in participants with and without data-entry experience. However, the question remains: what motivates checking in the first place?

Motivation has been highlighted in a theoretical discussion of human error analysis (e.g. Lourens, 1990) and it was suggested that if any model is to explain human error, it has to make explicit motivational as well as cognitive factors. The effect of motivation was examined by Skitka et al. (2000) in which participants were made to believe they were either accountable for their performance or not in flight-simulation tasks consisting of monitoring and tracking. Accountable participants were told that their performances were recorded and they would be required to explain and justify their actions. Non-accountable participants were told their performance data were not recorded and were not told anything about a post-experimental interview. The main finding suggests that accountable participants made fewer errors than non-accountable ones; and it was suggested that accountability made participants more attentive to their actions and more rigorous in their information seeking behaviour. Accountability can also have implications for an organisation's safety culture and its employers' attitudes towards errors (Dekker, 2009; Woods et al., 2010). If accountability involves blame and punishment, then people will consequently become defensive and unwilling to report their mistakes on the job. As a result, the organisation will not be able to learn from its mistakes.

Other forms of motivation have also been examined in a number of studies which looked at routine procedural errors. Instructional motivation has been found to have some effect in reducing certain procedural error but unable to completely eliminate it (Byrne and Davis, 2006). For example, Back et al. (2007) tested a punishment manipulation, which reset participants' scores in a computer game, and found that it was not effective in reducing a procedural error. The null effects of the punishment manipulations in these studies could be because of the impersonal nature of the manipulation. In other words, the punishment does not bear sufficient relevant (or personal) consequences to make it matter. In studying the effect of task interruption, Brumby et al. (2013) manipulated a time cost associated with errors in a routine procedural task: participants would be locked out and not be able to resume a primary task for a period of time (e.g. 20 s). It was found that participants made fewer errors as a result of the increased cost of making an error. This suggests that when errors have consequences that matter, participants respond to them.

Another way to impose consequences on task performance is to use a monetary incentive. For example, to test how well alternative fuselage designs facilitated evacuation in an emergency, Muir et al. (1996) offered a financial reward to the participants who managed to leave an aeroplane most quickly in a mock emergency situation. In basic psychological research, there is evidence that performance on a perceptual decision-making task is more affected by monetary incentives (earned points were converted into money) than symbolic incentives (earned points as indicative of performance) (Dambacher et al., 2011). Findings from EEG studies suggest that error-related negativity (ERN), a neural signal associated with error behaviour, was affected by monetary losses (Potts, 2011) and gains (Stürmer et al., 2011).

The effect of monetary reward was also tested in a computer based reaction time task (Wærsted et al., 1994), in which participants in a reward condition were told that good performance would earn them extra money; participants in a control condition did not receive such information. It was found that the reward condition led to improved performance in terms of faster reactions although the error rate did not differ from the control condition. In a recent study, the effect of monetary reward and time limit were tested in an auditory transcribing (hear-and-type) task (Lin and Wu, 2011). In the urgent condition, participants were instructed that they would be rewarded with extra payment for correct responses that were made within 600 ms. In the non-urgent condition, participants were paid a flat-rate regardless of task performance. It was found that although urgency led to improved typing speed, accuracy was compromised when compared to the non-urgent condition. This suggests that reward had a *partial* role in motivating fast task performance and the 600-ms time limit might have contributed to sacrificing accuracy. However, the design of the study does not allow one to draw definite conclusions regarding the effect of reward as it was compounded with time limit.

The objective of the work reported here was to examine the effects of reward and punishment under a single study because the two manipulations have not previously been concurrently tested. Specifically, the effects of monetary reward and punishment on checking behaviours will be compared in two experiments using a data-entry (or transcribing) task.[1] We focus our investigation on checking behaviour on a data-entry task because this type of task is common in many contexts: entering information into a library system, entering student marks from exam scripts into university system, entering items into an accounting system, entering data

[1] We will use the term data-entry and transcribing interchangeably as both terms refer to the same basic activity of copying information from one place to another.