Design Dimensions of Intelligent Text Entry Tutors

Per Ola Kristensson

Computer Laboratory, University of Cambridge,
JJ Thomson Avenue, Cambridge CB3 0FD, United Kingdom
pok21@cam.ac.uk

Abstract. Intelligent text entry methods use techniques from artificial intelligence to improve entry rates. While these text entry methods are useful in situations when a full-sized keyboard is impractical or unavailable, they also require substantial training investment from users. We hypothesize that intelligent text entry tutors may reduce this time and effort. However, before we set out to design these tutors we need to consider their design space. This paper contributes to this understanding by proposing and analyzing five design dimensions: automaticity, error correction, coverage, feedback and engagement.

Keywords: Text entry, intelligent text entry, text entry tutor, typing tutor.

1 Introduction

Intelligent text entry methods use techniques from artificial intelligence (AI) to improve entry rates. Examples of such methods are handwriting and speech recognition, the touch-screen gesture keyboard SHARK^2 [3] (commercialized as ShapeWriter/Swype/T9 Trace/Flext9), and the gaze writing method Dasher [10].

What these and other such methods have in common is that design restrictions, such as the form factor of the device or the capabilities of the user, reduce the input rate compared to ten-finger touch-typing on a full-sized keyboard. To compensate for a lower input rate intelligent text entry methods infer or predict what the user intends to write [2]. A challenge for some of these methods is that users need to relearn how to write using them, either completely (e.g. [10]) or partially (e.g. [3]). While commercial typing tutors are available for full-sized keyboards (e.g. Sega's *Typing of the Dead*), intelligent text entry methods pose unique challenges for learners. We here propose and analyze five design dimensions when building text entry tutors for them.

2 Design Dimensions

The first dimension is *automaticity*. For users to write fast they need saturate motor learning. This may require users to push themselves beyond an initial performance boundary or comfort zone. It has been suggested [4] that one effective way of achieving this is to use the expanding rehearsal interval algorithm [5] (also known as spaced repetition). This algorithm asks the user to write a word according to a certain rehearsal interval. If the user writes the word correctly within a set threshold the rehearsal

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 494–496, 2011.

interval is extended according to a multiplier. Otherwise, it is assumed the user has not reached automaticity for this word and the interval is left unchanged. Thus the algorithm regulates the tradeoff between rehearsing a word so often it is a waste of the user's time and rehearsing a word so seldom that the user never progresses. If users still fail to progress past their comfort zone, we also suggest investigating whether a series of immediate repetitions can trigger a transition.

The second dimension is *error correction*. Error correction is an unavoidable aspect of text entry and users need to know how to most effectively use the error correction techniques that are provided. For some interfaces, such as multimodal mobile speech recognition, there may be more than five different ways of correcting errors [9], each with its own pros and cons. The performance benefits in understanding when to use a certain error correction strategy can be substantial. For example, words that are out-of-vocabulary may never be correctly identified by the recognizer. In such cases expert users immediately fall back to another modality.

The third dimension is *coverage*. There are hundreds of thousands of words in a language. Fortunately two phenomena dramatically reduce this space. First, words tend to follow a highly skewed power law distribution known as Zipf's law[1]. It tells us that the most frequent words in a language comprise a large fraction of the text mass. For instance, it has been observed that around 46% of the British National Corpus consists of the 100 most frequent words [4]. Hence, substantial gains can be obtained by letting users practice only the top 100–200 most frequent words initially. Second, users have both an active and passive vocabulary. Passive words are the words we understand while active words are the words we as individuals use when we write and speak. While the former is in the order of tens of thousands of words the latter is usually only in the order of thousands. Hence, once a user is well practiced on the most frequent words in the language we suggest identifying the individual user's active words (e.g. by mining sent emails) and thereafter using these for further practice. It has been argued that the distinction between active and passive words is sharp rather than gradual [6] so dividing up the words users practice into these two sets is not as arbitrary as it may initially appear.

The fourth dimension is *feedback*. Here at least three factors need to be considered. First, users need to be informed on how they are progressing and they need to be rewarded for their progress (see also the fifth dimension *engagement* below). Second, the complexities in the underlying AI algorithms can result in behavior that puzzles users [2]. Ideally, text entry tutors can explain why AI algorithms fail to recognize or predict an intended word. For instance, Kristensson [2] describes various techniques such as confidence visualization and morphing the user's input into the recognized output to help users understand how systems process their data. Third, users who are using suboptimal strategies may benefit from immediate guidance. If empirical data on common misunderstandings among users is collected into an error library [7] then systems may be able to diagnose users' errors and provide remedial instructions [7].

The fifth dimension is *engagement*. Text entry tutors have to some extent explored this before, such as in a balloon game that explicitly stated "fun" as a design goal [4] and in a writing tutor for Japanese characters which was inspired by an existing game

[1] Zipf's law estimates the probability P_r of occurrence of a word in a corpus to be $P_r \propto 1/r^{\alpha}$, where r is the statistical rank of the word in decreasing order and α is close to one.

[8]. However, these tutors were relatively simple and repetitive. Users are generally impatient and need to be quickly convinced that learning a new text entry method is worthwhile. Therefore it is important to not only hook users initially but to also keep them hooked until they are able to use the new text entry method effectively. Clanton [1] discusses how user interface design can be aided by game design in this regard.

3 Conclusions

Teaching intelligent text entry methods poses unique challenges due to several factors, such as the complexities of the underlying AI algorithms, the need for users to quickly reach automaticity for frequently used words, and the need to hook users until they are able to use the new text entry methods effectively. To guide the design of intelligent text entry tutors we here proposed and analyzed five dimensions. These reflect design issues when teaching a wide array of text entry methods. We currently use these to guide our own development and hope they will stimulate further research.

Acknowledgments. This work was supported by the Engineering and Physical Sciences Research Council (grant number EP/H027408/1).

References

1. Clanton, C.: An Interpreted Demonstration of Computer Game Design. In: 16th ACM Conference on Human Factors in Computing Systems, Conference Summary, pp. 1–2. ACM Press, New York (1998)
2. Kristensson, P.O.: Five Challenges for Intelligent Text Entry Methods. AI Mag. 30(4), 85–94 (2009)
3. Kristensson, P.O., Zhai, S.: SHARK^2: A Large Vocabulary Shorthand Writing System for Pen-Based Computers. In: 17th Annual ACM Symposium on User Interface Software and Technology, pp. 43–52. ACM Press, New York (2004)
4. Kristensson, P.O., Zhai, S.: Learning Shape Writing by Game Playing. In: 25th ACM Conference on Human Factors in Computing Systems, Extended Abstracts, pp. 1971–1976. ACM Press, New York (2007)
5. Landauer, T.K., Bjork, R.A.: Optimum Rehearsal Patterns and Name Learning. In: Gruneberg, M., Morris, P.E., Sykes, R.N. (eds.) Practical Aspects of Memory, pp. 625–632. Academic Press, London (1978)
6. Meara, P.: A Note on Passive Vocabulary. Second Lang. Res. 6(2), 150–154 (1990)
7. Ohlsson, S.: Some Principles of Intelligent Tutoring. Instr. Sci. 14, 293–326 (1986)
8. Stubbs, K.: Kana No Senshi (Kana Warrior): A New Interface for Learning Japanese Characters. In: 21st ACM Conference on Human Factors in Computing Systems, Extended Abstracts, pp. 894–895. ACM Press, New York (2003)
9. Vertanen, K., Kristensson, P.O.: Intelligently Aiding Human-Guided Correction of Speech Recognition. In: 24th AAAI Conference on Artificial Intelligence, pp. 1698–1701. AAAI Press, Menlo Park (2010)
10. Ward, D.J., MacKay, D.J.C.: Fast Hands-Free Writing by Gaze Direction. Nat. 418(6900), 838 (2002)