Oops! Silly me! Errors in a Handwriting Recognition-based Text entry Interface for Children.

Janet C Read
Department of Computing
University of Central Lancashire
Preston
PR1 2HE
01772 893285
jcread@uclan.ac.uk

Stuart MacFarlane
Department of Computing
University of Central Lancashire
Preston
PR1 2HE
01772 893291
sjmacfarlane@uclan.ac.uk

Chris Casey
Department of Computing
University of Central Lancashire
Preston
PR1 2HE
01772 893278
ccasey@uclan.ac.uk

ABSTRACT

This paper describes an empirical study in which children aged 7 and 8 used handwriting recognition software and hardware to input their own unconstrained text into the computer. The children were observed using the software, and the behaviour of both the children and the system is described.

Handwriting recognition is a 'disobedient' technology; that is, it behaves erroneously, sometimes failing to generate correct representations of the child's intentions. This presents problems for the child, and these problems, and the strategies which the children adopted, are considered. Previous work on error correction with disobedient interfaces is used to provide grounding for the discussion.

Two models are proposed, one describing user-states, the second introducing the notion of 'tidal' error repair. These models are then used to suggest some strategies for the design of more usable handwriting recognition interfaces for children.

Keywords

Text Entry, Handwriting Recognition, Usability, Errors, Children

INTRODUCTION

This work is part of a larger study which is looking into the efficacy of handwriting recognition as a text entry method for young children. Young children seem particularly suited to handwriting interfaces. They have difficulties with the standard QWERTY keyboard, which can make text entry laborious and can cause them to lose their train of thought when using a keyboard as a composition tool (Read et al., 2000). There appears to be some evidence that children may write more fluently when using the handwriting tablet (Read et al., 2001a).

Research has indicated a shortage of work on the usability of handwriting recognition for this type of application.

Handwriting, speech and gesture recognition can be collectively termed 'disobedient interfaces'. This definition, first published in Snape et al. (1997), is used to describe recognition interfaces, which by virtue of their design have a propensity for error. These interfaces share certain characteristics; and work, which has been based on a study of one type of disobedient interface, can often inform the design of the others.

This paper begins with a discussion of some of the metrics which are used to evaluate usability with handwriting interfaces, referencing relevant work, and it then goes on to describe work which has been done, both with speech and handwriting, on error handling.

An empirical study is then described which looked at the patterns of errors in handwriting recognition with children.

Results from this study are then presented and discussed. The conclusion offers two models which are intended to inform design, and some suggestions are made for improvements in error handling.

USABILITY OF HANDWRITING INTERFACES

Previous studies on the usability of handwriting recognition have focused on the rate of recognition. Frankish et al., (1995) reported recognition rates as percentage correct figures and also recorded user satisfaction using questionnaires. The mean recognition rate in their study was 87% and correlations between recognition accuracy and user satisfaction were investigated. It was found that users were more satisfied with better rates of recognition.

MacKenzie and Chang, (1999) compared two handwriting recognizers by logging entry speed and accuracy. Accuracy was measured using percentage correct measures, and 87% - 93% accuracy was reported. In this test users copied words that were presented to them. Speed was expressed as wpm where 1word = 5 keystrokes or key taps. Users were given satisfaction questionnaires that similarly indicated that users were happier with high recognition rated processes.

A study by the authors used children as subjects and measured accuracy, speed and user satisfaction (Read et al., 2001b). In this case, characters per second were logged for time, CER (Character Error Rate) for errors and a number

NordiCHI 10/02 Århus, Denmark
© 2002 ACM ISBN 1-1-58113-616-1/02/0010…$5.00

of satisfaction metrics were used. This study compared handwriting and speech with obedient entry methods and by doing so, highlighted some of the problems with the traditional usability metrics. Recognition rates averaging 86% were reported for unconstrained text entry.

In each of these three studies, subjects were encouraged to ignore any errors that they noticed, and error correction was not included in the efficiency measures. The measures of recognition accuracy were therefore for uncorrected text.

Whilst these studies can give useful statistics, for most applications, users would need corrected text; therefore, the time and effort expended in discovery and correction is worth further investigation.

ERRORS

Donald Norman was one of the earliest researchers to consider the nature of human errors (Norman, 1981) and in later work; he categorized these by virtue of intention, offering descriptions of slips and mistakes. A slip was defined to be a human error where the intention was correct but there was an error made in carrying out the action, a mistake was an error in intention (Lewis and Norman, 1986). In the same paper, Lewis and Norman advocated that the designer should aim to minimize the incidence of errors, maximize their discovery and make recovery easy. Nievergelt and Weydert suggest that users of a system need to know the current system state, where they have previously been and what the possible alternatives are at the next interaction (Nievergelt and Weydert, 1980). In his early work on HCI, Booth wrote '*errors that occur at user interfaces are potentially one of the most useful sources of information*'. He went on to suggest that errors could be evaluated quantitatively (by counting) and qualitatively, with qualitative evaluation often offering illumination for design (Booth, 1989).

Errors in disobedient systems

In 'The Psychology of Human Computer Interaction', Card stated that '*the detection and correction of errors in a rule-based system is largely routine'* (Card et al., 1983). For a disobedient system, this is a clearly not the case. The nature of a disobedient system is such that both the system and the user can initiate errors.

Mankoff and Abowd have identified five key research areas for error handling of recognition-based interfaces. These are Error Reduction, Error Discovery, Error Correction techniques, Validation of techniques and Toolkit level support (Mankoff and Abowd, 1999). A later paper by the same authors considered user and system errors in a speech interface, developing a toolkit to assist in both the error discovery and repair (Mankoff et al., 2000). A study by Halverson was concerned with errors in speech recognition systems, and introduced the notion of 'cascading' errors; these being error repairs that lead to new errors (Halverson et al., 1999). This work was based on a paper in which the concept of 'error spirals' was introduced (Oviatt and VanGent, 1996). Error spirals are a feature of disobedient interfaces. A spiral is an analogy for the repeated attempts by the user that are often needed to fix an error. The depth of the spiral is defined to be the number of repeats needed before the recognition is correct.

Less work has been done on errors in handwriting than on speech. Errors that are likely to occur in handwriting interfaces have been categorized by Schomaker, (1994). These are described as; discrete noises, badly formed shapes, input that is legible by the human but not by the recognizer, badly spelt words, cancelled material and device generated errors. A subsequent investigation described three repair patterns, these being deletion, completion / insertion and overwriting (Huerst et al., 1998).

A study on text entry with children by Read et al., (2001b) classified the errors that may arise in disobedient interfaces when using text entry, in the following way; -

Spelling error - The child misspells the word.

Construction error - The child cannot form the letter or word correctly. In handwriting, 'a' may look like 'd'.

Execution error - The child fails to touch the tablet with the pen, or adds a spurious character.

Software induced error (hereafter referred to as **recognition errors) -** The software mis-recognises the word or character.

THE EMPIRICAL STUDY

Earlier work by the authors was carried out in a controlled experimental setting (Read et al., 2001b). While this gave useful information about efficiency and the overall effectiveness of the handwriting interface, it had some limitations. This work gave an indication that handwriting was as efficient as the keyboard; however, it did not consider the time spent on error repair.

For this new study, we looked at errors qualitatively with subjects using the interface in their own environment (Eason, 1984). Children aged seven and eight were given access to word processing software via an off-the-shelf handwriting recognition application and a standard QWERTY keyboard. This took place during their regular classes at school. Over the course of the study, which took several weeks, some children used both technologies and some used only one. The children were directed to either a laptop computer, which had the handwriting software on it, or a class computer, which had a standard keyboard attached, and the researcher subsequently observed them as they worked. The tasks carried out by the children varied from day to day, and from child to child. What was consistent was that each task involved free-text creation. That is, the child composed his or her own text at the computer, responding to a stimulus that had previously been supplied by the class teacher. The work was done as part of History, English and Religious Studies lessons.

Children worked in pairs, one child writing as the other assisted. This had the effect of increasing user dialogue and conversation, which in turn, encouraged free writing. It also had the effect of distracting the children from the observation process. Children were aware that their actions were being logged and this was also explained to them.

In each instance, about 50 'utterances' inputted by the user were recorded. This gave the children time to write a substantial chunk of text without putting too much pressure on them. Times were noted, which gave a crude measure of how efficient the interface was. Each time the child made an error, noticed an error, or corrected an error the subsequent actions of both the child and the interface were logged. The log sheet that was used had been developed during a pilot study with four children prior to the field trial. This sheet was used to record what the child wrote, what appeared on the screen, and what the child or system then did. Notes were made of interesting behaviour, illuminating conversations, and system failures. Each instance the child attempted to repair an observed error by writing something was considered an utterance.

Whereas both keyboard and pen entries were recorded in the field trial, this paper is concerned only with the observations of the activity at the pen interface.

RESULTS AND DISCUSSION

Error Discovery

For any error to be corrected it has first to be discovered. Typically, it is the user who is expected to find the error. This is not always easy, and for children, this will be more difficult than for adults. There is a correlation between the visibility of the error and the chance of it being detected. Spurious characters within words are more likely to be detected than incorrect but feasible letters. For instance, if a user writes 'Elodie' and the system generates 'Elo(ie' the chance of this being noticed is higher than if 'Elobie' is generated (Lewis and Norman, 1986). Visibility can also be enhanced by spell checking software that will generally highlight mis-spelt words. In this study, there was no spell checking software installed so children were not given this assistance.

The system can help the discovery of recognition errors by revealing some of its hidden information. Goldberg and Goodisman devised a probability of correctness system where the interface displayed the most likely choice of character in large font, with the two most likely alternatives displayed in smaller font below. This technique whilst appearing to help, actually failed to be of any use in user tests due to the high cognitive overheads (Goldberg and Goodisman, 1991).

Any discussion on the rate of error detection has to be from an unsteady base line. In this, and any other observational study, there may be some errors that would be missed by both the child and the researcher. There would also be some (probably very few) missed by the researcher but noticed by the child. It is only those errors that were noticed by the researcher that can be measured or considered.

In the observational study, almost 50% of the errors that the children missed were where the recognizer had changed the capitalization from the intended capitalization. There was no way of knowing whether the child had noticed a difference and considered it either not worth fixing or irrelevant, or whether the child had not seen the change. The remaining child-missed errors were of two types; spelling errors accounted for 32% of the total, and 22% were recognition errors. In these cases it was also impossible to know whether or not the child had chosen to ignore the error or missed it.

Error Recovery

Recovery is aided by clarifying the cause of the error; making the remedy easy to find, and by providing easy to use correction tools

When a child first noticed an error, he or she attempted one of the three repair strategies of deletion, completion/insertion or overwriting (Huerst et al., 1998). Depending on when the error was noted, children either rubbed out all or some of the word and then re-wrote, or, if they were part way through a word, they sometimes attempted to overwrite on, or scribble out the word using, the pen and graphics tablet. Those who tried this latter method, quickly realized the folly of it, and no child persisted with this technique. In general, words were rubbed out using either the backspace or the delete key on the keyboard.

The second attempt at the word or part-word was typically a new spelling (if the child had noticed a spelling mistake) or a repeat of a previous word or part-word (if the child assumed that the recognizer was at fault). As more attempts were needed children, who perceived the recognizer to be at fault, attempted some modification of style. If they found recognition impossible (and this seemed to happen for particular words) they sometimes rubbed the entire word, or even the whole sentence out, and wrote something different. In some cases, the child accepted the error and progressed without repairing it. There was evidence of increasing frustration as children got into spirals of erroneous outputs. Interestingly, children were visibly thrilled when after a long string of utterances; the recognizer finally 'understood' what they were writing!

Children had problems inserting punctuation. A common recovery strategy was to use the keyboard. It is worth noting that although a keyboard was available (and obviously working) no child attempted to use it to type letters in when the handwriting recognition failed.

Spaces were also problematic; the software introduced these itself when the pen was raised for a length of time. Many children just rubbed out the space using the backspace key and then continued. Children who realized that they got spaces by pausing mid-word were seen preparing the word in their head to ensure they could write it continuously, thus avoiding the space. Missing spaces were inserted using the space bar.

Some capital letters were difficult for the children to generate, and they were observed giving up attempting to get both capitalization and punctuation correct!

Watching the children, it was evident that some strategies were costlier than others in terms of time taken to repair errors. Some children always rubbed out whole words. Thus, they rubbed out letters that had been correctly recognized. This seemed to be a personal trait, as without exception individual children always either rubbed all or rubbed upto. Observing children using pen and pencil revealed a similar trend, with the same children using the same strategies for both paper and the handwriting interface. During the observations of the handwriting software, no child varied his style of erasing. Interestingly, when the same children used the keyboard for text entry, they also used the same erasing style as they had for the handwriting. The only difference was that some of the 'rub upto' children sometimes used the mouse to position the cursor and to then delete a single letter. The laptop computer, which the children used for the handwriting, had a touch pad for the cursor. This was unfamiliar to the children, and they did not use it. Consequently, any deletion involved rubbing backwards at least to the error.

The following table shows the progress of a child who rubbed all the word out, each time an error was spotted:

Table 1

Wrote	Got	Action	CER	Erased	Written
carried	warmed	Rubbed all out	3/7	6	7
carriedon	corrector	Rubbed all out	5/9	9	9
carr	cam	Rubbed all out	2/4	3	4
carred	caned	Rubbed all out	2/6	5	6
carred	(aired	Rubbed all out	2/6	6	6
carred	carrad	Rubbed all out	1/6	6	6
carred	(aor d	Rubbed all out	3/6	6	6
carread	carroad	Rubbed all out	1/7	7	7
carried	carried	Accepted	0/7		7
TOTAL	CCG= 7			EW=48+58= 106	

A COR (Cost of Repair) metric was used to illustrate the relative load caused by the two different erasing styles. This was defined to be the ratio of:

(The sum of the erased and written characters (EW)) / (Correct characters generated at the end of the spiral (CCG))

In the first example (Table 1) the COR was 106 / 7 = 15.1. That is, the child effectively rubbed out or rewrote the word 15 times! What is also interesting in this example is the fluctuation in the CER (Character Error Rate). It is evident that the generated word moved closer and further away from the desired representation during the repair. This phenomenon is discussed further in the conclusion.

This second example is from a child who chose to rub back to the error, rather than delete a whole word. Again, there is a spiral of repair attempts, but in this instance, the CER (as a metric based on the completed part of the intended word) is seen to fluctuate less.

Table 2

Wrote	Got	Action	CER	Erased	Written
centuary	cenioary	Rub out last 5	2/8	5	8
t	(r	Rub all out	2/4	2	1
t	t	Accept	0/4	0	1
uary	vary	Rub all out	1/8	4	4
uary	vary	Rub all out	1/8	4	4
au	all	Rub all out	2/8	3	2
uary	uary	Accept	0/8		4
TOTAL	CCG =8			EW =18+ 24 =42	

In this instance the COR is 42 / 8 = 5.3; significantly less than in the previous case.

Errors Avoidance

It has been suggested that users who understand the implications of their actions will make less errors (Lewis and Norman, 1986). If this is the case, it seems to be likely that children, as naïve users, will make more mistakes than adults.

Children adopted some interesting strategies for the avoidance of errors. Children reduced their own spelling errors by substituting words that they were unable to spell with other, easier, words. In some instances, the children asked one another, or the researcher, for spellings. Following a spate of mis-recognition children adapted their construction style and there was evidence that this adapted style was used in subsequent writing.

Children were seen learning how the recognizer worked; one child commented '*I'll have to work on my b's*' after the recognizer had struggled with her writing.

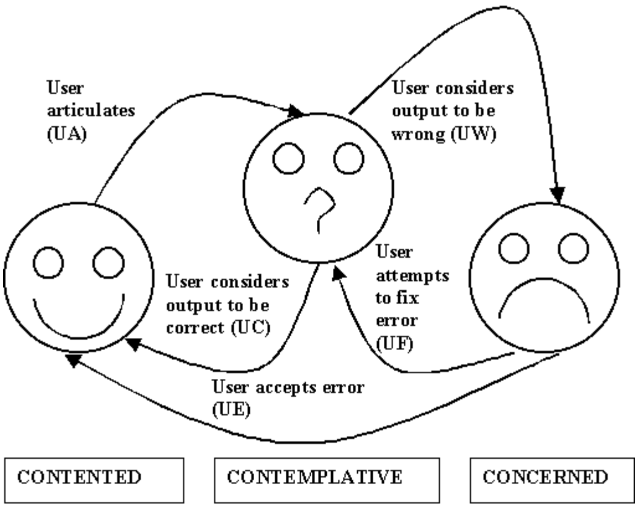
CONCLUSIONS

Two models have been developed which will assist the future development of handwriting interfaces for children.

Contented, Contemplative and Concerned Users

The user is engaged in a cyclic interaction with the interface. This sort of interaction was originally termed a recognize –act cycle by Card et al., (1983). We describe here a model of the three user states, which are, Contented, Contemplative and Concerned. These are modeled below.

Figure 1 – The CCC model.


User articulates (UA)
User considers output to be wrong (UW)
User considers output to be correct (UC)
User attempts to fix error (UF)
User accepts error (UE)
CONTENTED
CONTEMPLATIVE
CONCERNED

The challenge for the designer is to design systems which minimize the use of the UW / UF loop. When the users of a system are children, there may be a greater tendency to take the UE route, as children typically are used to both making, and having, mistakes in their work. This model is based on user perceptions – not on the state of the system, therefore, a user may take the UC route even when there is an error. This is more likely with children than with adults. It was gratifying to note that in the observational study, no child became over concerned about the system. It would not be sensible to pursue the development of an interface in which the state of the user was more often concerned than contented.

Tidal Error Modeling

As mentioned earlier in the paper; the practice of repairing an error can take the user further away from the correct representation as well as nearer. Given an efficient repair strategy, the user should be able to progress towards correct recognition, albeit it two steps forward, one step back at times.

This can be modeled using the analogy of an incoming tide. The tide will come in over a length of time. In the same way, well-intentioned user actions will move towards a correct representation over time. The number of waves needed to reach the correct representation would be equal to the 'spiral depth' as described by Oviatt and VanGent, (1996). Sometimes a wave may be exceptionally strong, and it may be then followed by weaker waves, thus giving the impression that the tide has turned and that the high tide (correct representation) has been missed. In a similar way, the user may interpret a badly recognized word as the worst possible (low tide) and then be surprised to find that it could get worse!

For the developer of disobedient systems, it is possible that there is no more absolute control over the errors than there is control over the waves. Certainly, it is the case that 'distance from correctness' – as described by the CER metric is not useful in determining what might happen next.

There is a clear case for enhancing the users' understanding of a system. Given a sound mental model of how a system works, users are better enabled to adopt strategies that will limit the occurrence of errors (Noyes et al., 1995). Providing a mental model for children, of a recognition process, is a challenge. In the same way that the strength of a wave cannot be predicted, many recognition errors seem difficult to predict, and therefore difficult to explain.

The user has to work with the recognizer, rather than against it. The authors of this paper are currently investigating the way that children make and recover from errors when using a pen and pencil in the hope that these natural strategies may be designed into a handwriting interface. Examples of natural strategies, which could be usefully incorporated, include the principle of overwriting and the pause, which often precedes a spelling in which the child has little confidence. This information could be used to assist in error discovery.

Giving children advice on 'cost-effective' erasing strategies, based on 'rubbing upto' rather than 'rubbing all', can speed up error repair. Whether or not there is much time to be gained in a position and delete strategy needs to be investigated. One of the problems with positioning as a technique for pen input is that the pen then has to act as both a pointer and a writing device, thus increasing the risk of mode errors.

What is important for both the children and the designers is to ensure that the user understands what is going on. The quote 'Oops! Silly me!' came from a seven year old child who was convinced that it was her fault that the word on the screen was not the word she had written!

References

Booth, P. (1989) *An Introduction to Human-Computer Interaction,* Lawrence Erlbaum Associates, Hillsdale.

Card, S., K, Moran, T., P and Newell, A. (1983) *The psychology of Human Computer Interaction,* Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Eason, K. (1984) Towards the experimental study of usability *Behaviour and Infromation Technology,* **3,** 133 - 143.

Frankish, C., Hull, R. and Morgan, P. (1995) *Recognition Accuracy and User Acceptance of Pen Interfaces*, Paper presented at ACM CHI'95,,503 - 510.

Goldberg, D. and Goodisman (1991) *STYLUS User Interfaces for Manipulating Text*, Paper presented at ACM UIST'91,,127 - 135.

Halverson, C., Horn, D. B., Karat, C. and Karat, J. (1999) *The Beauty of Errors: Patterns of Error Correction in Desktop Speech Systems*, Paper presented at Interact 99,,(Eds, Sasse, M. A. and Johnson, C.) IOS Press.

Huerst, W., Yang, J. and Waibel, A. (1998) *Interactive Error Repair for an Online Handwriting Interface*, Paper presented at CHI 98.

Lewis, C. and Norman, D., A (1986) In *User Centred Systems Design*(Eds, Norman, D., A and Draper, S. W.) Lawrence Erlbaum, Hillsdale, NJ, pp. 411 - 432.

MacKenzie, I. S. and Chang, L. (1999) A performance comparison of two handwriting recognizers *Interacting with Computers,* **11,** 283 - 297.

Mankoff, J. and Abowd, G. (1999) *Error Correction Techniques for Handwriting, Speech, and other ambiguous or error prone systems*, Paper presented at Interact 99.

Mankoff, J., Abowd, G. and Hudson, S. (2000) OOPS: a toolkit supporting mediation techniques for resolving ambiguity in recognition-based interfaces *Computers and Graphics,* **24,** 819 - 834.

Nievergelt, J. and Weydert, J. (1980) In *Methodology of interaction* (Eds, Guedj, R. A., ten Hagen, P., Hopgood, F. R., Tucker, H. and Duce, P. A.) North Holland, Amsterdam.

Norman, D., A (1981) Categorization of Action Slips *Psychological Review,* **88,** 1 - 15.

Noyes, J. M., Frankish, C. R. and Morgan, P. S. (1995) In *Personal Information Systems: Business Applications* (Ed, Thomas, P. J.) Stanley Thornes, Cheltenham, pp. 65 - 81.

Oviatt, S. and VanGent, R. (1996) *Error Resolution During Multimodal Human-Computer Interaction*, Paper presented at 4th Intl. Conference on Spoken Language Processing, Philadelphia,, Vol. 2 (Eds, Bunnell, T. and Isardi, W.) 204 - 207.

Read, J. C., MacFarlane, S. J. and Casey, C. (2000) *Where's the 'm' on the keyboard, mummy?* , Paper presented at Womens' Engineering Society, Preston, Lancs..

Read, J. C., MacFarlane, S. J. and Casey, C. (2001a) *Can Natural Language Recognition Technologies be used to enhance the Learning experience of Young Children?* Paper presented at Computers and Learning, Warwick, UK.

Read, J. C., MacFarlane, S. J. and Casey, C. (2001b) *Measuring the Usability of Text Input Methods for Children*, Paper presented at HCI2001, Lille, France,, Vol. 1 Springer Verlag, pp. 559 - 572.

Schomaker, L. R. B. (1994) *User-interface aspects in Recognizing Connected-Cursive Handwriting*, Proceedings of the IEE Colloquium on handwriting and Pen-based input,, Vol. number 1994/065 The Institute of Electrical Engineers, London.

Snape, L., Casey, C., MacFarlane, S. J. and Robertson, L. (1997) *Using Speech in Multimedia Applications*, Paper presented at TCS Conference, Bangor, Wales.