Opportunities and Challenges in Intelligent Mobile Text Entry

Per Ola Kristensson
Cavendish Laboratory
University of Cambridge
pok21@cam.ac.uk

CHI 2008, April 5 – April 10, 2008, Florence, Italy
Workshops and Courses: Usable Artificial Intelligence.

Abstract
Mobile phones are transcending into advanced multi-purpose devices capable of email, video and search. Even though mobile phones' technical capabilities are improving rapidly, mobile text entry remains a challenging research problem. A mobile text entry method must simultaneously offer a high text entry rate, low error rate, a short learning curve and a small form factor. Intelligent mobile text entry methods try to satisfy these often conflicting design dimensions using artificial intelligence (AI) techniques, such as pattern recognition and user modeling. This paper discusses several research opportunities and challenges in intelligent mobile text entry.

Keywords
Text entry, mobile text entry, intelligent mobile text entry, pattern recognition, artificial intelligence

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation: User Interfaces – *input devices and strategies, theory and methods*; I.5.2 Artificial Intelligence: Natural Language Processing – *language models*. I.5.5 Pattern Recognition: Implementation – *interactive systems*.

Introduction

Mobile text entry is a challenging research field. First, a mobile text entry method must be fast enough to let users comfortably write a larger text mass, such as an email. Second, it must also be accurate to be effective and thereby reduce user frustration. Third, since most users are not willing to invest significant training time to be accustomed to a new text entry method, a new mobile text entry method must also be easy to learn. For example, optimized keyboard layouts, while eventually quite efficient, require significant initial training investment to be efficient (*e.g.* [14]). Fourth, a mobile text entry method must have a small form factor. These are only a small subset of all design parameters that must be satisfied for a mobile text entry method to become successful. A full investigation, which identifies 22 critical design dimensions of mobile text entry, can be found in Chapter 6 in [4].

In summary, it is very difficult to create a new mobile text entry method that is both truly fast and practical. As a result, significant research effort has been invested in designing new text entry methods (see [4,9] for recent surveys of the mobile text entry field).

Can AI Contribute?

The fundamental problem of any text entry method – from speech recognition to a mechanical keyboard – is how to translate users' intentions into writing. Also, by minimizing users' effort in articulating text, text entry rate improves. AI researchers can contribute to this optimization process by observing properties in text entry that can be modeled and exploited by AI techniques. For example, because of language redundancy it is needless for users to completely spell out intended words when typing on desktop keyboards. In the case of the keyboard, human motor skills make up for the lack of bandwidth efficiency. However, when the keyboard is no longer appropriate, *e.g.* for a mobile device or a wall-sized display, language redundancy is a phenomenon that can unlock novel solutions.

Aside from handwriting and speech recognition, examples of AI contributing to text entry include predictive text such as the Reactive Keyboard [1] and Dasher [12], a novel abbreviated text entry method [10], and a graphical keyboard correction system [2].

Other recent examples are the elastic stylus keyboard (ESK) [4,6] and ShapeWriter (previously known as SHARK shorthand) [4,5,13]. The elastic stylus keyboard is an intelligent mobile text entry method that uses a pattern matching algorithm [4,6] to let users tap sloppier – and thereby potentially faster – than on a regular graphical keyboard (see Figure 1 for an example).

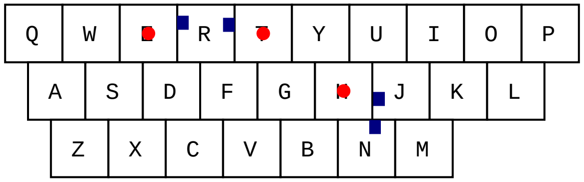

figure 1. The figure shows an example of a correction with the elastic stylus keyboard. The red circles indicate the center points of the word *the*. The blue rectangles show a user's actual landing points. By using a geometric pattern matching algorithm [6] the system can compare these sequences of points (in order) and return the word *the* instead of the verbatim result *rjnr*. The figure is taken from [3].

Opportunities

Intelligent mobile text entry opens up a new domain where algorithms, techniques and approaches from the AI community can be applied, such as commonsense reasoning [11], language modeling [1,2,8,10,12] and template matching [5,6].

In addition to algorithms, a largely unexplored territory is efficient visualizations and metaphors that teach users how AI algorithms work. Also, feedback that ensures users they are on track can be vital in some applications. As an example, [7] describes a system to enter commands (Copy, Track Changes, *etc.*) using a gesture recognizer [5]. The command interface shows a preview of the currently recognized command. Using this information, the user has an opportunity to decide whether to commit or cancel the command.

Further, intelligent mobile text entry methods also exploit the redundancies inherent in our languages. These redundancies are typically modeled in some form of language model. Other models can capture hand movement, typing speed, individual users, *etc.* For example, [2] corrects typing on stylus keyboards using a character-level language model and a key-press model. Language and user modeling can also be complemented with novel user interfaces, *e.g.* [8].

Challenges

The "Last Mile"

A clever algorithm does not guarantee a successful intelligent text entry method. Even though it may be possible to show theoretical quantitative advantages with certain approaches, there is always a risk that the user interface is cumbersome or that the algorithm is perceived as brittle by users.

An example is the work on designing the elastic stylus keyboard (ESK) [6]. The design of ESK followed an iterative design process. Two user studies were conducted using an early incarnation called the linear correction system. The primary difference between the linear correction system and the final ESK design was that the former could not cope with insertion and deletion errors (the user omitting a tap, or introducing a spurious extra tap). This deficiency resulted in a system that was perceived as brittle by users. For example, if a tap was accidentally omitted a completely different word would be outputted. Since users could not trust the system they chose not to take advantage of the technique. As a result text entry rate did not improve. This indicates that the "last mile" qualities of the AI algorithm can dramatically determine the end user benefits of the intelligent text entry method.

Providing High Performance User Experience

As hinted in the last paragraph, a slow text entry method is not a solution. In addition, text entry methods require high accuracy to prevent frustrated users. This makes text entry somewhat different than other AI applications, such as dialogue systems, where systems can recover from errors without necessarily impacting the overall user experience.

Working in Inflexible GUI Environments

Intelligent text entry methods often work on the word or sentence level. Currently most operating systems lack efficient editing-support for inserting and editing entire words in an application. In addition, text entry methods with a language modeling component may require the ability to read the text at the caret position of the focused window. These tasks are unfortunately either difficult or even impossible to implement in commonly used operating systems. This can limit the practicality of certain intelligent text entry methods.

Conclusions

Despite these challenges, intelligent mobile text entry methods have great opportunities to take advantage of advanced mobile devices with ever increasing memory and processing power. Whether users accept AI-inspired text entry methods is an open research question. Currently, there is a lack of rigorously collected longitudinal data (for a summary of empirical results in the mobile text entry field see Chapter 6 in [4]). To prevent solutions in search of problems, it is vital that researchers conduct experiments and field work to understand all the factors that affect actual text entry performance in practice.

In the end, if intelligent mobile text entry methods are to succeed, they must provide value to users – in the form of more comfortable, engaging and efficient text entry to the next generation mobile devices. This is an important goal: if mobile text entry methods do not improve, future text-entry dependent mobile applications risk being crippled, or scrapped altogether.

Acknowledgements

This research was funded by grants from Nokia, and Ericsson Research Foundation.

References

[1] Darragh, J.J., Witten, I.H. and James, M.L. The reactive keyboard: a predictive typing aid. *IEEE Computer 23*(11): 41-49, 1990.

[2] Goodman, J., Venolia, G.D., Steury, K. and Parker, C. Language models for soft keyboards. *Proc. AAAI 2002*, 419-424.

[3] Kristensson, P.O. Breaking the laws of action in the user interface. *Ext. Abstracts CHI 2005*, 1120-1121.

[4] Kristensson, P.O. *Discrete and Continuous Shape Writing for Text Entry and Control*. Doctoral dissertation, Linköping University, Sweden, 2007.

[5] Kristensson, P.O. and Zhai, S. SHARK^2: a large vocabulary shorthand system for pen-based computers. *Proc. UIST 2004*, 43-52.

[6] Kristensson, PO. And Zhai, S. Relaxing stylus typing precision by geometric pattern matching. *Proc. IUI 2005*, 151-158.

[7] Kristensson, P.O. and Zhai, S. Command strokes with and without preview: using pen gestures on keyboard for command selection. *Proc. CHI 2007*, 1137-1146.

[8] Kristensson, P.O. and Zhai, S. Improving accuracy in word recognizers using an interactive lexicon with active and passive words. *Proc. IUI 2008*, forthcoming.

[9] MacKenzie, I.S. and Soukoreff, R.W. Text entry for mobile computing: models and methods, theory and practice. *Human-Computer Interaction 17*: 147-198, 2002.

[10] Shieber, S.M. and Baker, E. Abbreviated text input. *Proc. IUI 2003*, 293-296.

[11] Stocky, T., Faaborg, A. and Lieberman, H. A commonsense approach to predictive text entry. *Ext. Abstracts CHI 2004*, 1163-1166.

[12] Ward, D.J., Blackwell, A.F. and MacKay, D.JC. Dasher: a gesture-driven data entry interface for mobile computing. *Human-Computer Interaction 17*, 199-228, 2002.

[13] Zhai, S. and Kristensson, P.O. Shorthand writing on stylus keyboard. *Proc. CHI 2003*, 97-104.

[14] Zhai, S., Sue, A. and Accot, J. Movement model, hits distribution and learning in virtual keyboarding. *Proc. CHI 2002*, 17-24.