Testing Predictive Text Entry Methods with Constrained Keypad Designs

Jun Gong, Peter Tarasewich

Human-Computer Interaction Laboratory
College of Computer & Information Science, Northeastern University
360 Huntington Ave, 202 WVH, Boston, MA, 02115 USA
{gjoliver, tarase}@ccs.neu.edu

Abstract

The problem of optimizing keypad designs for predictive text entry methods for handheld devices has received significant interest in recent years. The necessity of such optimization lies in the ever-increasing difficulty of inputting text into mobile devices, particularly those of extremely small size. Predictive keypad text entry methods, such as Tegic's T9, only require one key press for each letter of a word to be entered. However, such methods are ambiguous since one key sequence may correspond to multiple matching words because more than one letter is mapped to each key. Various keypad designs have been proposed in the past to optimize the placement of letters on keys to minimize ambiguous key sequences. In contrast to past research that allowed mapping of any letters to any keys, we hypothesize that maintaining alphabetical ordering across the keys of optimized keypad designs will result in higher novice usability and ease of learning, while still providing a more immediate benefit in terms of improved performance. Results from usability testing we conducted are encouraging and support our hypothesis.

1 Introduction

Accompanied by the growth of mobile electronics are the ever increasing difficulties of inputting text into such devices. Problems lie in the shrinking size of these devices and their screens, the constantly changing contexts in which they are used, and user attention that is often focused on multiple tasks (Gong & Tarasewich, 2004). However, text applications, such as text messaging (e.g. Short Message Service (SMS)), contact information lookup, and reminder lists, continue to be a major part of all mobile applications.

These observations warrant continued research into text entry methods for small devices. This paper presents the results of testing of an optimized eight-key keypad design for predictive text entry which keeps letters in alphabetical order against an optimal design that allows any letters to be placed on any keys but potentially offers higher user performance. The constrained (alphabetized) design is also tested against the standard telephone keypad design. The next section provides some background on predictive text entry with mobile devices. Section 3 presents our experimental methodology and describes how the optimal keypad designs were determined. Section 4 presents the results of our experiments and an analysis of the data collected. Section 5 provides a conclusion and areas for future work.

2 Background

The English language has 26 characters, while the normal mobile phone keypad only has 8 to 12 available keys. This means that in order to use a keypad for text entry, multiple letters must be assigned to the same key (as shown in Figure 1). Therefore, for text entry purposes, more than one key press is often needed to distinguish among the characters on each key.

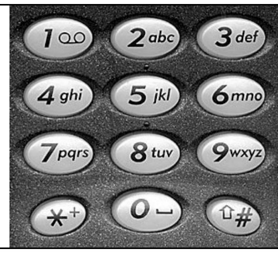

Figure 1: A standard mobile phone keypad with several letters assigned to each key

Dictionary based predictive methods, such as T9 from Tegic Inc., iTap from Motorola, and eZiText from Zi Inc., only require one key press for each character inputted. For example, to input the word "cat" on a standard mobile phone keypad, users press the key sequence "228". The key sequence is then matched to legitimate words stored in a dictionary. Ambiguity arises when more than one word matches a given key sequence. Under such conditions, usually the word with the highest frequency of occurrence will be chosen. If a different word is desired, it can be chosen from any remaining alternatives by pressing a special "next" key.

Recent research has looked at improving the performance of disambiguation algorithms. Masui (1999) developed a dictionary-based text entry method that uses the context of the phrase or document being typed. Rau and Skiena (1994) enhanced dictionary-based disambiguation by matching unknown words with the prefix or suffix of known words or using predefined rules to try to interpret the word. Hasselgren, Montnemery, and Svensson (2003) investigated a disambiguation algorithm that used probabilistic data from word pairs to guess a word based on the preceding word. MacKenzie, Kober, Smith, Jones, & Skepner (2001) developed a method called LetterWise that uses a stored database of prefix probabilities to guess the most likely letter of a key press based on the previously typed letters.

Besides creating efficient prediction algorithms, minimizing the amount of effort required to enter text using predictive methods can be achieved by designing keypads that result in the fewest predicted ambiguous key sequences for a given word corpus. Researchers have proposed various keypad designs, with different placements of English characters on each key (e.g., Lesher, Moulton, & Higginbotham, 1998; Saied, 2003; Hirotaka, 2003). Keypads can also be optimized based on various metrics, including the overall motor movement distance (Saied, 2003), the overall chance of having ambiguous words, and the overall number of key presses needed for inputting one character. Before our study, researchers looked for better keypad designs in an unconstrained manner, where any letters can end up grouped together on any key (e.g., "bfq" on one key and "ilv" on another). For instance, research such as Levine, Goodenough-Trepagnier, Getschow, & Minneman (1987), Kreifeldt, Levine, & Lyengar, (1989), Levine & Goodenough-Trepagnier (1990), and Lesher, Moulton, & Higginbotham (1998) described the optimization of character placement on various-sized keypads and the development of disambiguation algorithms using word dictionaries and statistics. Using such keypads for text entry, while eventually resulting in improved performance, would require a great deal of training and could result in much initial frustration. Therefore, we hypothesize that maintaining the alphabetical ordering of letters on keypads optimized for predictive text entry will increase novice user ease of learning and usability, and result in more positive user attitudes towards such methods, even through there may be tradeoffs in terms of predicted long-term text entry efficiency.

3 Methodology

Our research focuses on keypad design optimization for predictive text entry while supporting novice usability through the use of an alphabetization constraint. While potential long-term user performance may not be as great given this constraint, the similarity of our designs to current telephone keypads should greatly reduce any changeover costs, even for expert text entry users with current keypads.

In order to explore the above hypothesis, we conducted a usability test which compared the performance of 1) an eight-key constrained (alphabetized) keypad design, 2) an eight-key unconstrained design, and 3) the standard mobile phone keypad design. The optimal constrained keypad design was found through complete enumeration of all valid eight-key alphabetical keypad designs because the total search space was fairly reasonable (480,700 possible designs). The search space for the eight-key unconstrained design was realistically too large to search completely (approximately 1.6×10^{20} possible designs), so the eight-key unconstrained optimal keypad design was found using a Genetic Algorithm-based heuristic technique that we developed. We chose eight-key keypads for this testing because of the current popularity of using mobile phones to enter text messages. The telephone keypad design is a North American standard used on most mobile phones (Grover, King, & Kuschler, 1998). Our metric for evaluating the goodness of a particular keypad design was *disambiguation accuracy* (DA). DA is the probability that, if a sequence of keys is pressed, the "correct" word appears (that is, the word with the highest frequency of occurrence is the one intended). A higher DA value means fewer ambiguous key sequences, so an optimal design is one that maximizes the DA for a given list of words. The word list used in our study was a spoken word list derived from the British National Corpus (Leech, Rayson, & Wilson, 2001). Table 1 shows the best eight-key designs found for constrained and unconstrained keypads. It also shows the standard mobile phone keypad design.

Table 1: The best eight-key constrained and unconstrained keypad designs, and the standard mobile phone keypad design

Best Constrained 8-Key Keypad		Best Unconstrained 8-Key Keypad		Standard Mobile Phone Keypad	
ABCD	EFG	AFLX	GPUY	ABC	DEF
HIJKL	MN	MOQ	DEV	GHI	JKL
O	PQRS	BIT	NWZ	MNO	PQRS
TUV	WXYZ	HS	CJKR	TUV	WXYZ

3.1 Experiments

A usability experiment that compared text entry performance using the three keypad designs shown in Table 1 was conducted. We tested these three keypad designs using a simple dictionary-based disambiguation text entry system implemented on a Compaq iPAQ Pocket PC using .NET Compact Framework. A stylus was used to press virtual keys. An illustration of the three testing interfaces is shown in Figure 2.

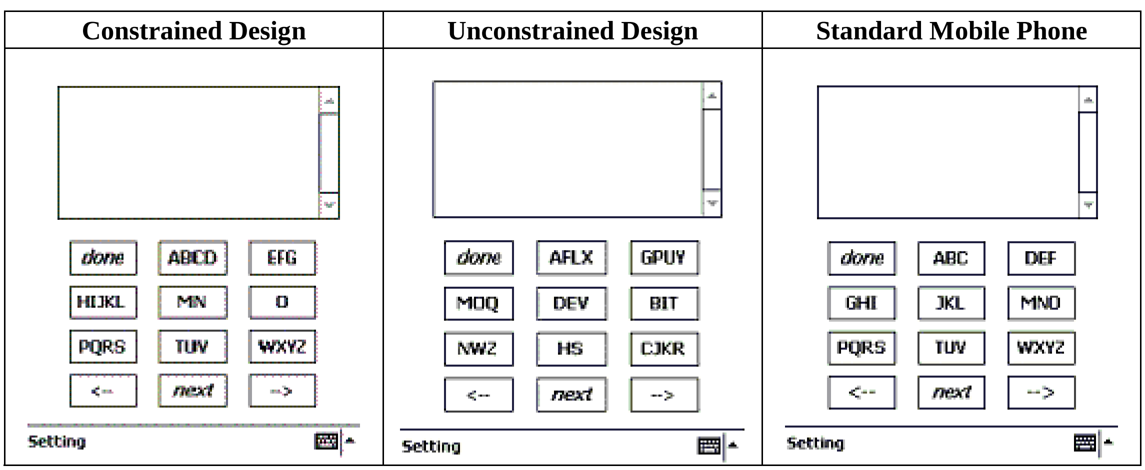

Figure 2: Three Testing Interfaces

Each interface consisted of a text box for showing the words entered and a simulated 12-button mobile phone keypad. Eight of the twelve buttons were used for characters. The other four were functional keys which enabled the users to accomplish basic text editing and multiple word entry as follows:

- "done" key: the sentence is complete, and the text box is cleared for the next sentence.
- "←" key: used to delete the last inputted character.
- "next" key: used to cycle through all the possible matching words if the key sequence is ambiguous.
- "→" key: used to commit the current word inputted, add a space, and start the next word.

Eight subjects, seven graduate students and one undergraduate student, voluntarily participated in the experiment over two sessions. Six of the subjects were male, and two were female. All but three of the subjects stated that they used a mobile phone daily. Only one subject did not carry a mobile phone at the time of the experiment, but did have prior experience using mobile phones. The median age of subjects was 29 years. Five subjects reported occasional use of text messaging, and two never sent or received text messages. Therefore, since none of the subjects used a mobile phone for text entry on a regular basis, they were all categorized as novice users for the purposes of this study. Subjects were not compensated for participating in the study.

Subjects were given questionnaires before the first testing session asking for background information. In the first session, subjects were asked to input six testing phrases using the constrained keypad design and another six phrases using the unconstrained design. The order in which subjects used the two interfaces was randomly determined, as was the order of the phrase sets. Subjects received training on the interface they used first, and were allowed to enter two practice phrases. After these two text entry tasks, the testing was repeated once more with the same two interfaces in the same order with two different sets of six phrases (so subjects entered 24 phrases in total – twelve on each interface). All of the testing phrases were obtained from a large testing database of sentences compiled by MacKenzie & Soukoreff (2003). Testing phrases contained only English characters and were carefully chosen so that each set contained the same number of ambiguous words for each keypad design. Example testing phrases are shown in Table 2. Subjects were told to work as quickly and accurately as possible, and to correct any errors that they noticed. Time and error rate were recorded for each phrase entered. After all four data entry tasks, subjects were asked for their opinions on the different keypad designs.

Session two, which occurred approximately a week later, was similar to session one except that subjects were asked to use the constrained keypad design and the standard mobile phone keypad design with the same four phrase sets. A questionnaire and training session was not administered. Subjects used the constrained design to enter the same two sentence sets they entered in session one. The remaining sets were used with the mobile phone keypad design.

Table 2: Example Testing Phrases

never too rich and too thin
the generation gap gets wider
gas bills are sent monthly
sad to hear that news
never mix religion and politics

4 Results and Analysis

Results, shown in Tables 3 through 5, are encouraging and support our hypotheses. The four columns show the average, standard deviation, minimum, and maximum data points for speed and error rate for each keypad design. "Speed" indicates the average speed (in words per minute (WPM)) that it took a user to input the text. The speed under Maximum and Minimum indicates the fastest and slowest average speed achieved by any one subject. The values in the "Error" columns indicate the average number of errors (corrected and non-corrected) users made for each word entered. For Maximum and Minimum, errors indicate the total number of errors by any one subject.

Table 3 below shows that speed with the constrained keypad design was significantly higher (t=0.000) than that with the unconstrained design (6.74 WPM versus 5.50 WPM), while these two designs seem to have similar error rates and standard deviations.

Table 3: Comparing the Speed and Error Rates of the Alphabetically Constrained Keypad Design and the Unconstrained Keypad Design

Design	Mean		Std Deviation		Maximum		Minimum	
	Speed	Errors	Speed	Errors	Speed	Errors	Speed	Errors
Constrained	6.74	0.39	20.7	0.62	14.6	0	2.67	3
Unconstrained	5.50	0.31	19.0	0.64	11.3	0	2.30	4

For the subjects tested, average text entry speed using the alphabetically constrained keypad design did not significantly differ (t = 0.169) from that of using the mobile phone keypad design (7.89 WPM versus 8.22 WPM) as shown in Table 4. Error rates were also approximately equal. Of note, however, is that the standard deviation of errors made with the constrained keypad is somewhat higher than with the mobile phone keypad, suggesting that some users did not adjust completely to the constrained mobile phone keypad design. This may have changed if the design was used more over time.

Table 4: Comparing the Speed and Error Rates of the Alphabetically Constrained Keypad Design and the Mobile Phone Standard Keypad Design

Design	Mean		Std Deviation		Maximum		Minimum	
	Speed	Errors	Speed	Errors	Speed	Errors	Speed	Errors
Constrained	7.89	0.47	27.9	0.81	16.2	0	4.48	4
Standard Tel	8.22	0.42	29.3	0.63	17.4	0	4.17	3

Furthermore, a significant increase in text entry speed (t = 0.000) was observed between sessions one and two for the constrained keypad design (Table 5), suggesting that the constrained design is fairly easy to learn and remember. But it is interesting to note here that the error rate and standard deviation for the second session are higher than in the first session. One possible explanation for this may be that users were more used to the constrained keypad design during the second session, and started to make careless mistakes with increased entry speed.

Table 5: Comparing the Speed and Error Rates of the Alphabetically Constrained Keypad Design Across Two Testing Sessions

Design	Mean		Std Deviation		Maximum		Minimum	
	Speed	Errors	Speed	Errors	Speed	Errors	Speed	Errors
Session 1	6.74	0.39	20.7	0.62	14.6	0	2.67	3
Session 2	7.89	0.47	27.9	0.81	16.2	0	4.48	4

5 Conclusions and Future Work

Our research has contributed to the knowledge about predictive text entry techniques through the design of constrained keypads. We hypothesized that an optimized alphabetically constrained keypad design would have performance comparable to the standard mobile phone keypad design, but would allow better novice ease of learning and usability compared with an unconstrained keypad design with potentially better performance. The results of our experiments support these hypotheses, and suggest that an optimized constrained keypad designs may be a suitable text entry option for certain types of users and applications.

Our research discussed in this paper is just the beginning of the design and testing of constrained predictive keypad designs. Future works may include evaluating the performance of constrained keypad designs of other sizes (e.g. 3-button or 5-button keypads). Since numbers and punctuations were not considered in this experiment, they will be included in future experiments. We are also working on finding a better dictionary, such as a common SMS word list, to improve the predictive accuracy of the text entry system under more realistic text entry conditions in the mobile environment. Finally, we think it is possible to use predictive next letter highlighting to enhance the predictive text entry system. Predictive next letter highlighting is a technique that enables the text entry system to predict several most probable next letters based on linguistic models, and highlight them on the keypad so that users will be able to find the keys easier with less search effort.

6 References

Dunlop, M.D., & Crossan, A. (1999). Dictionary based text entry method for mobile phones. *In Proceedings of the Second Workshop on Human Computer Interaction with Mobile Devices*.

Gong J., & Tarasewich P. (2004). Guidelines for handheld mobile device interface design. *In Proceedings of the 2004 DSI Annual Meeting*.

Grover, D.L., King, M.T., and Kuschler, C.A. (1998). Patent No. US5818437, Reduced keyboard disambiguating computer. Tegic Communications, Inc.

Hasselgren, J., Montnemery, E., & Svensson, M. (2003). HMS: A predictive text entry method using bigrams. *In Extended Abstracts of 10th Conference of the European Chapter of the Association of Computational Linguistics* (pp. 43-49)

Hirotaka, N. (2003). Reassessing current cell phone designs: using thumb input effectively. *In Extended Abstracts of CHI '03*.

Kreifeldt, J.G., Levine, S.L., & Iyengar, C. (1989). Reduced keyboard designs using disambiguation. *In Proceedings of the Human Factors Society 33rd Annual Meeting* (pp. 441-444)

Leech, G., Rayson, P., & Wilson, A. (2001). *Word frequencies in written and spoken English: Based on the British national corpus*, Pearson ESL.

Lesher, G.W., Moulton, B.J., & Higginbotham, D.J. (1998). Optimal character arrangements for ambiguous keyboards. *In IEEE Transactions on Rehabilitation Engineering 6*, 415-423.

Levine, S.H., Goodenough-Trepagnier, C., Getschow, C.O., & Minneman, S.L. (1987). Multi-character key text entry using computer disambiguation. *In Proceedings of the RESNA 10th Annual Conference* (pp. 177-178)

Levine, S.H., & Goodenough-Trepagnier, C. (1990). Customised text entry devices for motor-impaired users. *Applied Ergonomics 21(1)*, 55–62.

MacKenzie, I.S., Kober, H., Smith, D., Jones, T., & Skepner, E. (2001). LetterWise: Prefix-based disambiguation for mobile text input. *In Proceedings of the 14th annual ACM Symposium on User Interface software and technology*, (pp. 111-120)

MacKenzie, I.S., & Soukoreff, R.W. (2003). Phrase sets for evaluating text entry techniques. In *Extended Abstracts of CHI 2003* (pp. 754-755)

Masui, T. (1999). POBox: An efficient text input method for handheld and ubiquitous computers. *In H.-W. Gellersen (Ed.) Handheld and Ubiquitous Computing, First International Symposium* (pp. 289-300)

Rau, H., & Skiena, S.S. (1994). Dialing for documents: An experiment in information theory. *In Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology* (pp. 147-155)

Saied B. N. (2003). A fast, full-text entry system for small electronic devices. *In Proceedings of the 5th International Conference on Multimodal Interfaces* (pp. 4-11)