An Evaluation of Mobile Phone Text Input Methods

Lee Butts and Andy Cockburn
Human-Computer Interaction Lab
Department of Computer Science
University of Canterbury
Christchurch, New Zealand
{lrb34, andy}@cosc.canterbury.ac.nz

Abstract

The rapid growth of Short Message Service (SMS) text messaging is generating substantial commercial and research interest in fast and efficient text input methods for mobile devices. This paper presents an empirical study that compares three mobile phone text input techniques. The methods are 'multi-press input with timeout', 'multi-press input with a next button', and 'two-key'. The study shows that there is a significant speed difference, in words per minute (wpm), between the methods. The multi-press with next method provided the most rapid text input at 7.2 wpm, followed by multi-press with timeout at 6.4 wpm. The two-key method was the slowest at 5.5 wpm. These results are much slower than those predicted by Fitts' Law models reported in prior research. Subjective results regarding learnability, errors and efficiency showed no significant difference between the methods.

Keywords

Text input, mobile phone, keypad input, text messaging.

1 Introduction

The use of text messaging is growing every month. Between January and December 2000, Short Message Service (SMS) use grew from 4 billion to 15 billion messages per month (Figure 1), an increase of *375%* in one year. The GSM Association predicts 25 billion messages per month by December 2001 with 200 billion in total for 2001 (GSM Association Press Release 2001).

Mobile phones, however, are not naturally suited to text input. The standard (ISO/IEC 9995-8 1994) layout of a mobile phone uses 12–15 keys to allow basic text input (Figure 2). These 12–15 keys must cover the 26 letters of the alphabet and all of the punctuation and numerical characters. For this reason each key is overloaded so that, for example, the 6 key is mapped to M, N and O. Additional special characters and punctuation may also be assigned to each key.

This overloading creates a need for methods that allow the user to specify which letter on a particular key they want when it is pressed. For example, if a user presses the number 2, the mobile phone does not know whether the user wants an A, B or C. Several methods have been implemented to solve this disambiguation problem. The following section discusses related work in the area of text input with mobile devices. Then three input methods used in this study are discussed, followed by details of the experimental design. Results are then presented, discussed and compared to model predictions, along with discussions on further work and the conclusions.

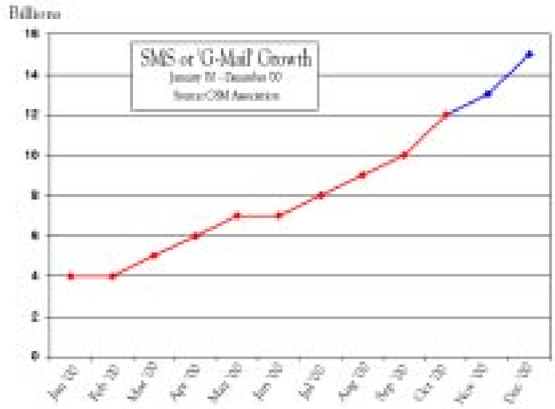

Figure 1: World-wide growth of SMS use.

2 Related Work

Silfverberg, MacKenzie & Korhonen (2000) propose a mathematical model based on Fitts' Law for predicting the speed of text input methods. Their paper notes that designing an input method for a mobile phone is a labour intensive task, and that empirically evaluating new methods is time-consuming and expensive. They therefore proposed a predictive model that allows rapid comparison of text-entry interfaces, and estimates text-entry rates in words per minute. Their predictions for multi-press with timeout, multi-press with a next button, and two-key are compared to the results of this study later in the paper.

James & Reischel (2001) undertook a study using a similar method to that used in our study. They compared *predictive text input* as proposed by Dunlop & Crossan (2000) to the multi-press method. They do not specify which variation of the multi-press method was used (multi-press with next button or multi-press with timeout, as described in the following section). They compared their results with the modelled predictions of James & Reischel (2001), and showed slower text entry rates than predicted. They state that the predictive model is based on a level of expert performance that rarely exists in real users.

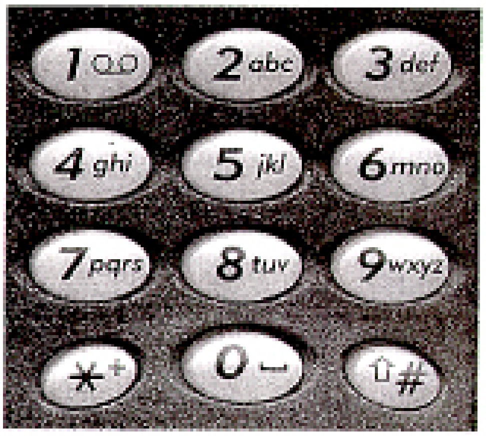

Figure 2: A standard ISO 12-key keypad as found on the Nokia 5110.

3 Text Input Methods

The core problem with entering text on a mobile phone is that each key is mapped to three or four different letters. A multi-press method works by cycling through letters on a key with each successive press. For example, one press of the 2-key enters an 'A', a second press of the 2-key changes the 'A' to a 'B' and a third to 'C'. However, this causes problems when two letters on the same key are entered. The sequence 222 may mean the user wants to enter 'ABC' or it can also be interpreted as 'C'. Two variations of multi-press that solve this problem are used in this study. The third technique analysed in this study uses a combination of two key presses for every character input. Each of these techniques is described below.

3.1 Multi-press with timeout

The multi-press with timeout technique uses a fixed timeout to decide when a user has finished cycling through letters on a key. Once the user presses a key a timeout starts and if the same key is pressed before the timeout expires (usually 1–1.5 seconds), the interface will cycle through the letters available on that key. For example, to enter 'ABC', the user must press the '2'-key, then wait for the time-out to expire. They then press the '2'-key twice, where the two presses are separated by less than the timeout interval. Once the timeout has expired, they must press the '2'-key three times where the time gap between each pair of presses is less than the timeout. In other words, 'ABC' can be entered using the key sequence 2-22-222 where a dash signifies waiting for the timeout to expire.

3.2 Multi-press with next button

The multi-press with next button technique replaces the timeout with a 'next' button. Instead of waiting between successive letters, the user presses the next button to signify that they have finished cycling through letters on that key. To enter the string 'ABC' the user would have to press the following key sequence: 2<next>22<next>222.

3.3 Two-key

The two-key method takes a substantially different approach. Instead of cycling through letters on a key, two key presses are used, where the first press indicates the desired key and the second press identifies the position on that key. For example, the letter 'N' is on key '6' at the second position, so the user would press '62'. Likewise, a 'G' would be entered using '41' ('4'-key, first position).

This technique means that every letter takes exactly two presses to enter, unlike multi-press where the number of presses ranges from one to three (or more for punctuation). Two-key, however, is not suited for entering punctuation or special characters, as the user needs to be able to see all the letters mapped to each key in order to determine the position of the key they desire.

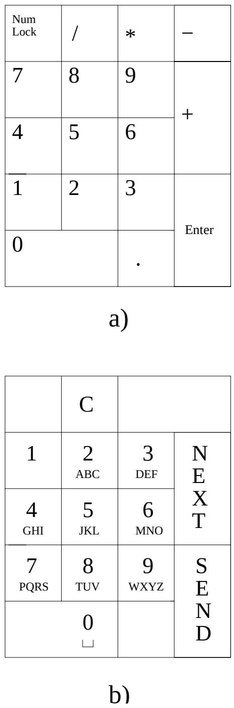

Figure 3: The numerical keypad before a) and after b) re-labelling.

4 Experimental Design

The aim of the evaluation was to determine whether there were reliable differences between the efficiency of the three text-entry techniques. The experimental design was a repeated measures one-factor analysis of variance. The single factor was 'text entry interface' with three levels: multi-press with timeout, multi-press with next, and two-key.

4.1 Subjects

Eight participants were used in this pilot study. All were male post-graduate Computer Science students. Although the number of subjects in this study is small, the repeated-measures design of the experiment reduces the impact of individual variability on the data analysis. Three of the subjects had no prior experience with text messaging, three subjects sent between one and five messages per week, and two sent more than five messages per week. Each of the subjects was trained in the use of the techniques prior to the experimental tasks.

4.2 Procedure

Each subject was asked to enter five sentences using each input method. They used the numeric keypad of a keyboard relabelled as shown in Figure 3. The

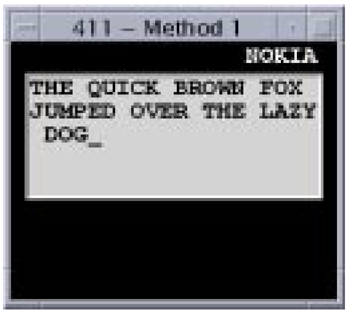

Figure 4: The mock mobile phone interface.

No.	Sentence	Chars.
1	hi joe how are you want to meet tonight	39
2	want to go to the movies with sue and me	40
3	what show do you want to see	28
4	we are meeting in front of the theater at eight	47
5	let me know if we should wait	29

Table 1: The five test sentences used.

subject's input appeared on screen via a simple emulator written in Tcl/Tk. Figure 4 shows the emulator containing some example text. Subjects were asked to practice with each input method until they felt comfortable using it. Usually this training period was extremely short, with most of the subjects (including those without prior experience) reporting that they felt familiar with the interface after less than one minute of training. The training data was not included in the logged data file.

There were three versions of the emulator, one for each input method. Each emulator recorded the elapsed time from the first keypress to the 'Send' key. The timer was reset so that the time taken between sentences was not recorded. The subject number, input method, elapsed time and entered text were written to a file after each sentence.

The sentences used were taken from James & Reischel (2001) with the aim of allowing direct comparison of results for the different methods tested. The sentences are listed in Table 1. They are conversational sentences intended to mimic real use of text-messaging. Most subjects, however, commented that the sentences were too long, not realistic and tedious to enter. The use of mobile phones for text input has led to a new language of abbreviations such as 'r' instead of 'are' and 'c u l8r' instead of 'see you later'. Experienced users suggested how they would have entered the sentences using such common abbreviations. James & Reischel (2001) also use a set of 'newspaper' sentences. These were long sentences similar to ones found in a newspaper article. These were not used in this study because they are not representative of mobile phone use.

For consistency with prior experiments, word per minute rates were calculated from an assumed 5.98 characters per word (Dunlop & Crossan 2000).

The order in which each subject used the three methods was counter-balanced to minimise the impact of learning effects.

After the user had completed a set of five sentences, they were asked to complete three questions regarding the learnability, error-rate and efficiency of the input method they had just used. The Likert scale questions (1 disagree, 5 agree) are shown in Table 2.

Having completed all five sentences with all three input methods, the subjects were given the option to modify their previous answers. Several subjects commented that this was useful because their impression of a certain method had changed after using one of the others.

Each subject's participation in the experiment lasted approximately 30 minutes.

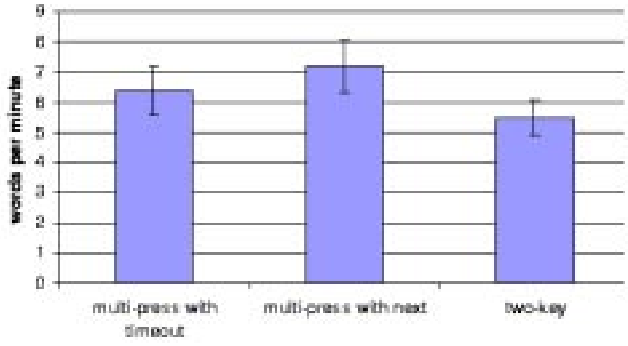

Figure 5: Mean text-entry rates per method. Error bars show one standard error above and below the mean.

5 Results

Most subjects found the tasks tedious and frustrating. This is a well-known 'feature' of text messaging. The choice of sentences may have increased subject frustration, with many complaints about the length of the sentences, the absence of 'appropriate' text-messaging abbreviations, and American spelling.

5.1 Performance Results

The mean speed (in words per minute) for the multi-press with timeout, multi-press with next button, and two-key were 6.4 (s.d. 2.3), 7.2 (s.d. 2.4) and 5.5 (s.d. 1.7), providing a reliable difference: F(2,14)=12.12, $p < .01$. Figure 5 depicts these results for the three techniques.

5.2 Subjective Measures

The subjective ratings for each interface type with regard to learnability, error-rate and efficiency are summarised in Table 2. There was no significant difference between the three methods with regard to learnability (Friedman Test, $\chi_r^2 = 1.31$, $df = 2$, $N = 8$, $p = 0.52$), error-rate (Friedman Test, $\chi_r^2 = 1.56$, $df = 2$, $N = 8$, $p = 0.46$) and efficiency (Friedman Test, $\chi_r^2 = 2.69$, $df = 2$, $N = 8$, $p = 0.26$).

5.3 General Observations

Several subjects appeared to have strong habits developed from prior experience with text messaging. Some complained that they found it hard using a method other than the one found on their mobile phone. Subject 4, for instance, suggested an improvement to the multi-press with next button method that was supported by his phone. The suggestion was to be able to press the 'next' button twice to enter a space character.

Two subjects (Subjects 3 and 7) commented that they thought their performance with the two-key method would improve if they used it regularly. Subject 7 also believed that over time one would start to learn the two digit code for each letter which would increase text input speed. In contrast, Subject 4 complained that the two-key method required too many key presses.

Question	MP - Timeout	MP - Next	Two-key
Q1. I found this method easy to learn	4.3(0.5)	4.1(0.8)	3.8(1.0)
Q2. I did not make many mistakes with this method	3.9(0.8)	4.3(1.0)	3.5(1.3)
Q3. Overall, this input method was efficient to use	3.4(0.9)	3.8(1.0)	3.0(0.9)

Table 2: Mean (standard deviation) responses to 5-point Likert scale questions.

5.4 Confounding Factors

Use of an Emulator
The fact that a computer based emulator was used instead of an actual mobile phone was regrettable, as discussed in Section 5.4. Users had to enter text using a keyboard and shift their eyes from the keyboard to the screen, which is a much greater distance than a user would usually have to deal with using a real mobile phone. Using a keyboard is also different to holding a mobile phone in one's hand. The keyboard may have increased the speed of text input due to the large size and slightly different layout from a mobile phone keypad. However, the extra time needed to look up at the screen will have reduced the calculated speeds of each method. Although the use of a keyboard is not ideal, the same keyboard was used for all input mechansisms, and there is no reason to believe that it will have a different impact across any of the three input methods.

Number of Subjects
Although only eight subjects were involved in this pilot study, the results provide strong initial evidence of the relative efficiencies of the different text entry techniques.

Test Sentences
Most subjects considered the test sentences to be unrealistic. The subjects complained that when text-messaging radical abbreviation is standard practice.

The sentences were used in order to reproduce some of the results found by James & Reischel (2001). In further studies we will investigate abbreviated test sentences.

6 Comparison with Related Work

Table 3 shows the comparison between the predictions from Silfverberg et al. (2000), the results for multi-press from James & Reischel (2001) and the results from this study. The predictions calculated using the two-handed, index finger method (reported by Silfverberg et al.) have been used as this is the closest approximation to the technique used by subjects in this study. The speeds calculated from this study are similar to the empirically determined values of of James & Reischel (2001), but substantially lower than the predictions of Silfverberg et al. (2000). We agree with James & Reischel (2001) that the model proposed by Silfverberg et al. (2000) predicts speed according to a level of expertise that is not realistic. Users of the next button method will soon realise the prediction of 22.72 words per minute is extremely optimistic.

7 Further Work

Our further work will focus on broadening the range of techniques evaluated. In particular, we wish to compare the performance of the new generation of 'predictive' text entry systems with the multi-press with next input scheme. Current predictive input systems include T9[1], which is found on many commonly used phones such as the Nokia 3210, LetterWise[2] and WordWise. Predictive text messaging systems aim to increase the efficiency of text entry to a one keypress per character level. They do so by comparing the words that a seqeunce of keypresses can represent and guessing which one was intended depending on word probablilities. For example, the key sequence 2-2-5-3 (see Figure 2) can represent *able, cake, bald* or *calf.* Predictive methods suggest words in order from the most to least likely, as recorded in dictionary of words and statistical information on their probability.

We are also interested in conducting longitudinal studies that allow us to inspect the evolution of user performance with text-messaging over weeks and months of frequent use.

8 Conclusions

This study compared three alternative schemes for supporting text input on mobile phones. Results show a reliable difference between mean text entry rates. The multi-press with next button technique allowed the most rapid text entry, with a rate rate of 7.2 words per minute. Multi-press with timeout achieved a rate of 6.4 wpm and two-key allowed text entry at 5.5 wpm. Interestingly, the experimental participants found all three techniques frustrating and cumbersome to use. Subjective ratings of the three techniques did not yield significant differences, but the multi-press with next attained the highest mean rating, followed by multi-press with timeout and two-key. By comparing our results with Silfverberg et al.'s mathematical model that aims to predict text-entry performance, our study supports prior conclusions that although a valuable tool to interface designers, mathematical models can be misleading if not compared to actual performance James & Reischel (2001).

References

Dunlop, M. D. & Crossan, A. (2000), 'Predictive text entry methods for mobile phones', *Personal Technologies* pp. 134–143.

GSM Association (2000), 'Membership statistics'. `www.gsmworld.com/membership/mem_stats.html`

GSM Association Press Release (2001), *More than 200 Billion GSM text messages forecast for full year 2001*, GSM Association, Avoca Court, Temple Road, Blackrock, Co. Dublin, Ireland. `www.gsmworld.com/news/press_2001/press_releases_4.html`

ISO/IEC 9995-8 (1994), *Information systems - Keyboard layouts for text and office systems - Part 8: Allocation of letters to keys of a numeric keypad*, International Organisation for Standardisation.

James, C. L. & Reischel, K. M. (2001), Text input for mobile devices: comparing model prediction to actual performance, *in* 'Proc of CHI2001', ACM, New York, pp. 365–371.

[1] T9 is a Registered Trademark of Tegic Communications.

[2] LetterWise and WordWise are a registered Trademarks of Eatoni Ergonomics.

Paper	Method		
	MP - Timeout	MP - Next	Two-key
Silfverberg et al. (2000)	22.5	27.2	25.0
James & Reischel (2001)	7.98 (variation not specified)		N/A
This study	6.4	7.2	5.5

Table 3: Comparison with model predictions and related work.

Silfverberg, M., MacKenzie, I. S. & Korhonen, P. (2000), Predicting text entry speed on mobile phones, *in* 'Proc of CHI2000', ACM, New York, pp. 9–16.

Welch, B. B. (1997), *Practical Programming in Tcl and Tk*, 2 edn, Prentice Hall.