

A Study of Variations of QWERTY Soft Keyboards for Mobile Phones

Justin Cuaresma, I. Scott MacKenzie

Dept. of Computer Science and Engineering, York University
Toronto, Ontario, Canada M3J 1P3
cse93146@cse.yorku.ca; mack@cse.yorku.ca

Abstract - Three recent Qwerty soft keyboard variations are studied: Curve (equivalent to Swype), T+ (equivalent to SureType), and Octopus (equivalent to the BlackBerry Z10 keyboard). In an experiment with 12 participants, the Octopus keyboard surpassed the standard Qwerty keyboard by the 4th phrase entered, reaching an entry speed of 70 wpm on the 9th phrase. The standard Qwerty soft keyboard had a mean entry speed of 54 wpm. The Curve is shown to be the least efficient of the Qwerty variations with a mean entry speed of 35 wpm. It is also the most error prone of the four keyboards. The T+ keyboard is shown, through the power law of learning, that it can surpass the standard Qwerty keyboard in entry speed after about 19 phrases of input.

Keywords: Soft keyboards, Qwerty, mobile phones, Swype, predictive text entry, T9

1. Introduction

Mobile phones are becoming powerful computers, sized to fit in one's hand. What before was a device for calling and texting is now a smartphone with embedded technologies, such as cameras, light sensors, vibro-tactile actuators, accelerometers, gyroscopes, GPS receivers, and so on. Today's uses for smartphones extend well beyond calling and texting, and include the capability to write and send e-mail and even type out notes and complete documents using a note app or word processor.

However, such technological progress has also caused a change in traditional practices. A common trend among many smartphones is the removal of the physical keypad, a feature that was common to all mobile phones just a few years ago. Many smartphones today employ touchscreen technology. The devices use on-screen "soft" keyboards. Many of these soft keyboards resemble the classic and renowned Qwerty keyboard layout. However, new issues arise when using soft keyboards. The fact that there are over 26 on-screen buttons to fit into a limited screen size is a design challenge (Sears and Zha, 2003). Common criticisms relate to the sensitivity of the touchscreen, the button size, the efficiency of the layout, etc.

There are other keyboard layouts besides Qwerty. Physical keyboard variations, such as Dvorak, reduce finger fatigue by placing the most predominantly used letters on the home row of the keyboard. Opti is an optimized soft keyboard layout, with frequent letters positioned in the centre, infrequent letters in the perimeter (MacKenzie and Zhang, 1999). It is demonstrably more efficient than a Qwerty soft keyboard after a succession of learning trials. The original empirical comparison found that the Opti keyboard surpassed the input speed of the standard Qwerty layout after the tenth session, which was equivalent to about four hours of practice. The reason is that performance with a Qwerty layout tends to level off, suggesting that people's ability to type faster is bounded by the Qwerty letter arrangement. However, designers are hesitant to employ alternate keyboard layouts due to the large social bias favouring Qwerty. Employing alternative layouts requires users to invest time learning a new input style. In today's competitive marketplace, "different" is of limited appeal to consumers. Our research seeks to examine soft keyboards that use the Qwerty letter arrangement while including novel tricks to optimize and enhance input.

To combat the stagnant efficiency of the Qwerty layout on soft keyboards, designers and software engineers have adapted a different strategy. Following the motto, “if it ain’t broke, don’t fix it,” modern keyboard designs often retain the Qwerty letter arrangement while implementing unique interaction techniques. One of these new techniques includes a notion called “Shape Writing,” first introduced by an application called “Swype” on Android devices. To enter text while “swyping,” users draw the word in one continuous motion without lifting the finger. An evaluation found that the Android Swype keyboard performed just as well as a regular tapping and provides a better experience for users (Nguyen and Bartha, 2012).

Another common feature to enhance text entry is predictive text. Of predictive text entry is not new; it has been used on mobile phones since the early 1990s with a feature commonly known as “T9” (Silfverberg et al. 2000). T9 was designed to reduce the keystrokes required for text entry on a phone keypad (which initially used multi-tap). Using the algorithm behind T9, keyboards can be designed with different letter-to-key mappings (Kreifeldt et al. 1989). One example is the “reduced Qwerty” keyboard, which positions letters on a single row of keys, while maintaining the Qwerty letter arrangement. There are only nine letters keys. Letters are arranged vertically: QAZ on the first key, WSX on the second key, EDC on the third key, and so on (Green et al. 2004). In an empirical evaluation, users nearly matched the speed of typing on a standard Qwerty keyboard, despite the need to disambiguate key presses (like T9 on a traditional phone keypad).

Blackberry Pearl devices use a Qwerty layout on a physical keyboard, but place two letters on each key: QW, ER, TY, etc. The keyboard is called “SureType,” and requires disambiguation, much like T9. Essentially, SureType is T9 disguised in a Qwerty layout.

In soft Qwerty keyboards, predictive text works by auto-completing or auto-correcting a user’s input. An example of predictive text in modern smartphones is the Apple iPhone’s auto-correct feature. When active, the phone automatically corrects spelling errors to the closest possible word the user is entering. The recently released Blackberry Z10 includes a new keyboard feature which displays predicted words on top of each subsequent key. The idea of using predictive text with different interaction methods is a continuing trend in the smartphone industry (Inverso et al., 2004).

The goal herein is to compare modern tweaks or variations of the Qwerty keyboard. Our evaluation compares Qwerty variants and assesses whether they are faster or less prone to errors. The keyboards included are the standard Qwerty, Octopus, TouchPal Curve, and, TouchPal T+. These are presented in the next section.

1.1. Qwerty Variant Keyboards

Qwerty variants maintain the underlying Qwerty style, while incorporating some deviation from the original (e.g., swipe versus tap). Before presenting our evaluation methodology, the four keyboards are described. The first is the standard Qwerty keyboard on the iPhone. See Fig. 1. To study the effect of predictive text in all variations, the auto-correct feature of the standard iPhone 4 keyboard was enabled.

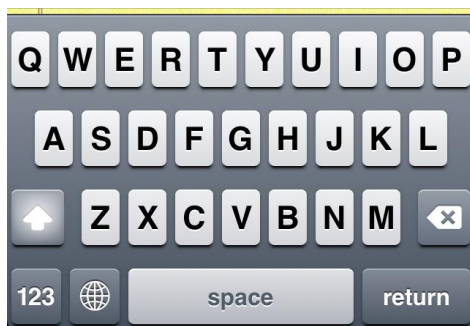


Fig. 1. Standard Qwerty keyboard.



Fig. 2. Octopus keyboard.

The second variant is called “Octopus”, implemented by K3A (<http://ok.k3a.me/>). This Qwerty variant mimics the functionality of the keyboard on the new Blackberry Z10. That is, as letter keys are tapped, predictions are shown on top of the next letter. To choose the prediction, the user swipes up on the letter, thus autocompleting with the suggested word. The keyboard was installed via Cydia (homebrew app store) with a license fee of \$5 USD. See Fig. 2.

The third variant is the “Curve” developed by TouchPal (www.touchpal.com). This style mimics the well-known Swype keyboard discussed above. It uses swiping gestures wherein users do not lift their finger until all the letters of a word are traced. The tracing does not have to be precise, as an algorithm is used to determine the word based on the shape of the gesture. See Fig. 3.



Fig. 3. TouchPal Curve keyboard (input of “brown”).



Fig. 4. TouchPal T+ keyboard.

The fourth variant is called “T+”, also developed by TouchPal. This variant mimics the SureType input method used in previous generation Blackberry Pearl devices. It acts much like T9 on a Qwerty layout. Each button contains two letters, while maintaining the Qwerty spatial arrangement of letters. In addition, each button requires just a single tap which is accompanied with a predictive algorithm to anticipate the word the user is trying to enter. See Fig. 4. The TouchPal app, which consists of both Curve and T+ layouts, can be downloaded from the App Store for free.

2. Method

We conducted an empirical evaluation to compare the standard Qwerty soft keyboard with the three variants described above.

2.1. Participants

The experiment included 12 participants. Due to the standardization of the standard Qwerty keyboard layout on all modern computers and devices, every participant had prior experience with the standard Qwerty keyboard layout thus, introducing a slight bias in the experiment. Since there were four keyboards, participants were divided into 4 groups to counterbalance the order of testing and offset learning effects. Each group contained 3 participants. There were 6 males and 6 females. The mean age of participants was 20.7 years ($SD = 2.5$ years). Six of the participants owned an Apple iPhone, three owned an Android device, two owned a Blackberry (with a physical keyboard) and one owned a regular mobile handset (non-smartphone). Participation was based on candidate willingness and there was no incentive offered for participation.

2.2. Apparatus

The evaluation included a questionnaire with 14 items. The first nine items (completed before testing) pertained to general details such as age, gender, and type of phone. The last five items (completed after testing) pertained to opinions on the different variations of Qwerty keyboards experienced during the experiment. The questionnaire was generated using Google Forms, a subcomponent of Google Drive.

Testing was performed on an Apple iPhone 4 with custom firmware installed (to allow external 3rd party plugins).

2.3. Procedure

The participants were invited to take a seat at a desk where a computer and an iPhone were presented to them. Each participant was then briefed on the purpose of the experiment. Using the computer, the questionnaire was presented and the participant was asked to complete the first nine items. Upon completing the initial items, the participant was shown the iPhone with the first Qwerty variant opened, prepared for entry. The participant was briefed on the features of that particular keyboard, including a quick visual demonstration. The participant was then given the phone and asked to enter the phrase, *the quick brown fox jumps over the lazy dog*. This was repeated 8 times (9 phrases, total). The next three variants were then tested in the same manner (and in order according to the counterbalancing).

Participants were instructed to use both hands as well as both thumbs where applicable. There were no practice periods given. The participants were asked to use and learn the features of the given variant. Data were collected manually using a stopwatch and through visual inspection of the text entered. The data were entered in an Excel spreadsheet to calculate summary results and build charts. An analysis of variance was performed using a free downloadable Java application call Anova2 (<http://www.yorku.ca/mack/HCIbook/>). Upon completion of the 36 iterations (nine for each variant), the participant was then asked to complete the remainder of the questionnaire.

2.4. Design

The experiment was a 4×9 within-subjects design. The independent variables were keyboard variant (4 keyboards) and phrase iteration (9 phrases). The dependent variables were entry speed and error rate. Speed was calculated in words per minute (wpm) using Eq. (1). The error rate was calculated by dividing the number of characters in error by 43 (the number of characters in the test phrase). All data were recorded manually, as noted above. The order of administering the keyboard variants employed a 4×4 balanced Latin square to offset learning effects. There were four groups for each order and each group was comprised of three participants.

$$\text{Speed} = \frac{\left(\frac{43}{5}\right)}{\left(\frac{x}{60}\right)}, \text{ where } x = \text{recorded completion time in seconds} \quad (1)$$

The total amount of input was 4 keyboards \times 9 phrase iterations \times 12 participants = 432 phrases.

3. Results and Discussion

The effect of group (order of testing) was not statistically significant both for entry speed ($F_{3,8} = 0.593$, ns) and for error rate ($F_{3,8} = 0.497$, ns). Thus, we conclude that counterbalancing had the desired effect of offsetting learning effects due to the order of testing. In the following sections the results for entry speed and error rate are given, along with models confirming the power law of learning.

3.1. Entry Speed

The grand mean for entry speed over all 432 phrase iterations was 45.7 wpm. The Octopus keyboard averaged 54.7 wpm which was 1.4% faster than the standard Qwerty keyboard at 54 wpm. See Fig. 5. The T+ keyboard averaged 38.7 wpm, about 9.6% faster than Curve at 35.3 wpm. The Octopus and standard Qwerty are both approximately 40% faster than both T+ and Curve. The large difference between the standard Qwerty and Octopus keyboards, and the Curve and T+ keyboards we attribute mainly to the participant's lack of experience with Shape Writing as well as to their lack of experience with a T9-like system in a touchscreen context. Not surprisingly, the differences in entry speed by keyboard variant were statistically significant ($F_{3,24} = 17.95$, $p < .0001$). A Bonferroni-Dunn post hoc test revealed that all pairwise comparisons were statistically significant except standard Qwerty vs. Octopus and Curve vs. T+ ($p < .05$). This result is apparent in Fig. 5.

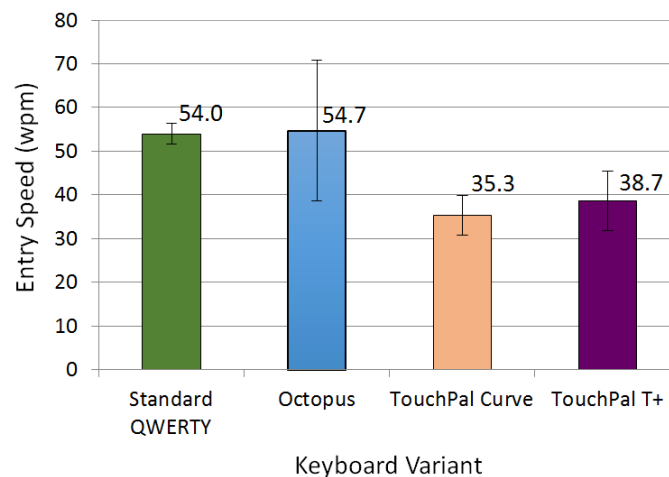


Fig. 5. Entry speed (wpm) by keyboard variant. Error bars show ± 1 SD.

Although the standard Qwerty and Octopus keyboards performed similarly overall, an examination of the learning progression over the 9 phrase iterations reveals a different outcome. As seen in Fig. 6, on the first iteration, Octopus had the slowest entry speed. This was likely due to visual demand, since participants did not have prior experience with an Octopus or similar keyboard. The Octopus does not suggest the desired words at first. Participants were observed to first look around the keyboard and then continue to type regularly if the word did not show up as a suggestion. Nonetheless, we see a remarkable 187% increase in entry speed for the Octopus keyboard from the first iteration to the last. Overall, the effect of phrase iteration on entry speed was statistically significant ($F_{8,64} = 29.0$, $p < .0001$).

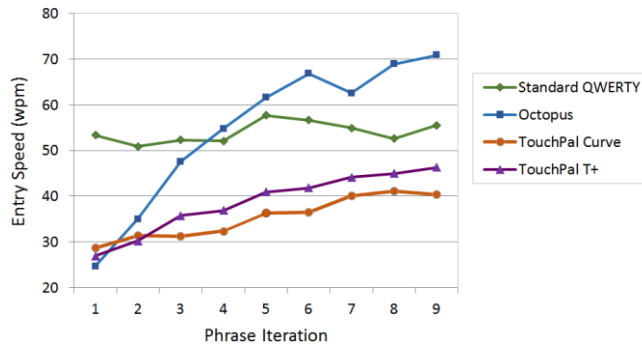


Fig. 6. Entry speed (wpm) by phrase iteration and keyboard variant.

Fig. 6 also shows the Curve having a 40% increase from the first to last iteration. T+ improved by 72% in entry speed by the 9th iteration. In contrast, the standard Qwerty keyboard only improved by 4% by the 9th trial. However, as previously stated, the design of the experiment gives Octopus a particular bias, since there is only one phrase that is entered 9 times. Octopus is both predictive and adaptive. As words are revisited, they migrate to the top of the candidate list and appear sooner on next-letter keys. Words like “the” appear immediately with the first key tap. Others, like “quick,” eventually appear after the first key press (in this case, above the “u” after “q” it tapped). Over the course of the 9 iterations, participants began to notice this behaviour and exploited it to improve their performance. In fact, by the 9th phrase iteration, the nine words in the phrase could be entered with nine swipes. Since repeatedly entering the same phrase is unlike typical text entry, it is evident the superior performance with the Octopus keyboard is in part due to the experimental protocol. Nevertheless, it does appear that the entry speed with the Octopus keyboard can at times exceed 60 wpm (see next section). The Octopus keyboard is reinstalled after every participant session to clear the phrase history. Participants struggled with the Curve the most as they had trouble tracing the letters for various words and this slowed their entry speeds.

3.2. Power Law of Learning

Over the 9 phrase iterations, significant learning was observed. To explore this, the power law of learning was used to model learning and to forecast into the future by 15 additional trials. See Fig. 7.

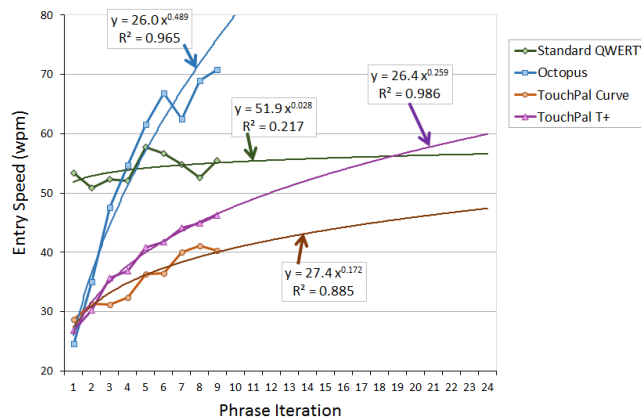


Fig. 7. Power law of learning for standard Qwerty, Octopus, Curve, and T+, with extrapolation to the 24th phrase.

Clearly, the models for the Octopus and T+ keyboards are good predictors as the squared correlations are above .95. The model for the Curve is also good with a squared correlation near .9. In contrast, the model did not provide a good predictor for the standard Qwerty keyboard ($R^2 = 0.217$). This result is similar to Qwerty predictors in other research (MacKenzie and Zhang, 1999) and is attributed to

the simple fact that participants are well along the learning curve for the standard Qwerty keyboard when they show up for testing. (It is sometimes joked that participants “cheated for ten years” before testing!)

Fig. 7 suggests that the Octopus keyboard is faster than the standard Qwerty keyboard by the 4th trial and can go well beyond 75 wpm. However, this prediction is not likely to bear out for random text, since there is less opportunity for adaptive behaviour. The extrapolation also shows that in comparing the standard Qwerty and T+ keyboards, a crossover point is reached at about 18 trials, after which the entry speed for T+ is expected to exceed that for the standard Qwerty keyboard. T+ may reach about 60 wpm on the 24th trial. The models also illustrate that the Curve keyboard would only reach about 47 wpm by the 24th trial; hence, even with practice, the Curve keyboard may continue to be slower than the other variations.

3.3. Error Rate

The grand mean for error rate was 4.0%. The Curve keyboard was the most error prone at 5.4%, while the Octopus keyboard had the lowest error rate at 1.9%. See Fig. 8. Common errors for the Curve keyboard consisted of participants tracing over an extra letter, which then resulted in a completely different word. The low error rate for the Octopus keyboard was in part due to the adaptive nature of the design. Once a word is entered, it is saved in the keyboard’s history. If, for example, a participant correctly enters “quick”, then on the next iteration, “quick” appears on top of “u” after entering “q” and the participant need only swipe up on “u” to complete the word. However, this feature can also backfire. If the participant incorrectly spells “quick” (e.g., “qiuck”), then the entire word must be re-entered correctly, letter by letter, before any form of adaptation emerges. Overall, the variance in the data was sufficiently high that the effect of keyboard type on error rate was not statistically significant ($F_{3,24} = 2.32$, $p > .05$).

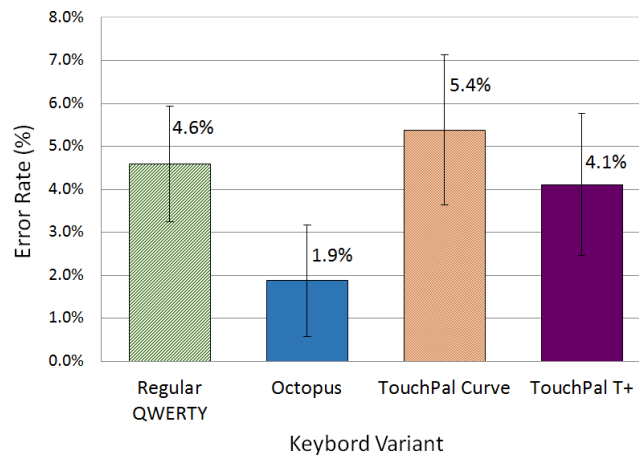


Fig. 8. Error rate by keyboard variant. Error bars show $\pm 1 SD$.

Although there was an overall decrease in error rates from the 1st phrase (5.9%) to the 9th phrase (4.5%), there was considerable variability. This yielded a non-significant statistical effect of phrase iteration on error rate ($F_{8,64} = 1.86$, $p > .05$).

3.4. Participant Feedback

The questionnaire included five items soliciting participant feedback on the keyboards. Participants were split on their choice of preferred keyboard. The preferences for a particular keyboard numbered as follows: 2 for standard Qwerty, 4 for Octopus, 3 for Curve, and 3 for T+. On least preferred keyboard, the Curve led with 6 dissatisfied participants. Comments here included “inaccurate”, “hard to use”, “I hate swiping”, “annoying and ugly”.

4. Conclusion

We conducted an experiment comparing a standard Qwerty soft keyboard with three design variants: Octopus, Curve, and T+. The Octopus keyboard was the fastest and also the least error prone, demonstrating an ability to reach 70 wpm while maintaining a low error rate of just 2.0%. Through the power law of learning, the T+ keyboard was shown to have potential to surpass the entry speed of the standard Qwerty soft keyboard after about 24 phrases of entry. It is also shown that the least efficient Qwerty variant is the Curve having both the slowest entry speed and the highest error rate. Even with an extrapolation to 24 phrases, the Curve is still expected to be slower than the standard Qwerty counterpart.

When participants were asked about keyboard preferences, the results showed that the Curve keyboard was the least preferred. One participant commented on their choice of least preference, stating “during the experiment, some of the letters... were too far from each other and feel as if it took more time to swipe back and forth between them.” The Octopus, on the other hand, was the most preferred Qwerty variant. A participant explained his choice of preference stating, “[Octopus] learns my typing pattern over time and when it does it only takes one keystroke to achieve the desired word.”

It is encouraging to see that designers and engineers are not forcing consumers to learn a more efficient – yet new and strange – keyboard design. Instead, slight modifications and improvements in the Qwerty keyboard have arrived as a by-product of social bias towards Qwerty. The Octopus (an implementation of the Blackberry Z10 keyboard) preformed best in our evaluation. It was also well liked and this was at least partly because it does not deviate too far from the original Qwerty interaction style.

This research is open to further effort. Some issues to consider for future work include increasing the number of phrase iterations as well as assigning phrases randomly from a collection so that all Qwerty variations are rendered equivalent.

Acknowledgements

Special thanks to Steven Castellucci for providing guidance on empirical research methods.

References

- Inverso, S. A., Hawes, N., Kelleher, J., Allen, R., Haase, K. (2004). Think and Spell: Context-Sensitive Predictive Text for an Ambiguous Keyboard Brain-Computer Interface Speller, “Biomedizinische Technik 49 Suppl.,” Dublin, Ireland, pp. 53-54.
- Green, N., Kruger, J., Faldu, C., St. Amant, R. (2004). A reduced Qwerty keyboard for mobile text Entry, “Extended Abstracts of the ACM Conference on Human Factors in Computing Systems,” New York, ACM, pp. 1429-1432.
- Kreifeldt, J. G., Levine, S. L., Iyengar, C. (1989). Reduced Keyboard Designs Using Disambiguation, “Proceedings of the Human Factors and Ergonomics Society Annual Meeting,” San Diego, CA, HFES, pp. 441-444.
- MacKenzie, I. S., Zhang, S. X. (1999). The Design and Evaluation of a High-performance Soft Keyboard, “Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,” New York, ACM, pp. 25-31.
- Nguyen, H., Bartha, M. C. (2012). Shape Writing on Tablets: Better Performance or Better Experience? “Proceedings of the Human Factors and Ergonomics Society Annual Meeting,” San Diego, CA, HFES, pp. 1591-1593.
- Sears, A., Zha, Y. (2003). Data Entry for Mobile Devices Using Soft Keyboards: Understanding the Effects of Keyboard Size and User Tasks. *International Journal of Human-Computer Interaction*, 16, 163-184.
- Silfverberg, M., MacKenzie, I. S., Korhonen, P. (2000). Predicting Text Entry Speeds on Mobile Phones, “Proceedings of the ACM Conference on Human Factors in Computing Systems,” New York, ACM, pp. 9-16.