Increasing QWERTY Efficiency with Predictive Color Cues

Mason D'Arcy,
Sachin Khargie,
Kai Xu
Dept. of Electrical Engineering and Computer Science
York University
Toronto, Ontario, Canada M3J 1P3
masond@my.yorku.ca
sachin.khargie@gmail.com
kevxu@my.yorku.ca

ABSTRACT

Text entry using soft keyboards is a significant bottleneck in user interfaces. QWERTY has become the defacto standard. We have integrated a next-letter prediction mechanism into a QWERTY soft keyboard by coloring keys as a function of probability. We then compared two different color prediction palettes to a non-colored keyboard, in order to test for any accuracy or input speed discrepancies. The results were a significant reduction of speed when using the color cue system (15.6% slower) and a small increase in accuracy (2.27% faster). Six participants entered a common set of phrases into an android application that prototyped the system, and their accuracies and speeds were measured. Users remarked that distractibility from the color cue system may account for their speed loss.

Keywords

Soft keyboard, predictive keys, QWERTY, android, typing speed, typing accuracy, color cues

INTRODUCTION

The QWERTY keyboard layout was designed over a century ago with little thought towards efficiency of text input. The layout of QWERTY was determined to prevent jamming in typewriters. Typewriter keys were placed far apart if they were commonly typed in sequence. [7] In spite of not being designed for speed, QWERTY became recognized in 1971 by the International Standards Organization. It is largely perceived as the de facto standard keyboard layout for computer interfaces, and dominates hard and soft keyboards to this day. [8]

There have been successful attempts to create a more efficient keyboard layout than QWERTY. For example, Norman and Fisher [7] showed the Dvorak keyboard layout has approximately a 5% advantage in performance over QWERTY for expert users. Zhai et al. [10] confirmed the CHUBON, FITALY, and OPTI keyboards outperform QWERTY. With respect to learning, alphabetically organized keyboards have an initial advantage over non-alphabetically organized keyboards for novices. [7] Alphabetically organized keyboards become similar in efficiency to non-alphabetically organized keyboards as the typist becomes proficient. [7]

Although some keyboards layouts have been shown to outperform QWERTY, it has yet to be dethroned. Thus, the industry is stuck with QWERTY and the bottleneck of user text-input in computer interfaces. The number of devices and users inputting text has massively increased in recent years, making finding methods of faster text input an area of focus and interest. To increase throughput, there have been a slew of features, tools, and augmentations created to enhance or be used in tandem with QWERTY.

One of the major tools to increase text-entry speeds and accuracy has been predictive systems. [3] These systems are varied and widely used, enabled as a default setting in many desktop, iOS, and Android environments. [3] Many modern soft keyboards have an interface component called a "suggestion bar" where either fully completed words, corrected spellings, or phrases are suggested. [3]

While the suggestion component is useful for efficiency, Arnold et al. [3] suggests that these kinds of predictive systems have an effect on the content that users write. Predictive text suggestions are taken as cues, and may cause users to write short, predictable language. [3] Most experiments regarding these systems only focus on words per minute and accuracy using transcription studies. The effect on the content that users write is not often studied, instead only the speed and accuracy of their input is measured. [3] One idea to avoid this issue, while still leveraging the power of a predictive system, would be to integrate the suggestion component directly into a soft keyboard itself. Since QWERTY is dominant, it seems a fitting candidate for the layout choice.

A method to integrate the suggestion component into a soft keyboard is by dynamically mutating attributes of each key. Instead of a discrete phrase completion component, each key will change color on a spectrum between red and green. This color change would be a function of how likely a key is to be the next stroke. Using a natural language processing model, a prototype will map the probability of each letter to a hexadecimal color value in order to aid a user.

The predictive color strategy is meant to improve the accuracy and text input speed of users on a QWERTY soft keyboard, without polluting the content of what the user is inputting. Thus, a comparison will be drawn between a regular QWERTY soft keyboard, and a QWERTY soft keyboard with dynamic coloring. The related work section describes studies focusing on similar techniques and tools meant to increase user text input speed. The method section gives an overview of the predictive coloring apparatus, procedure, and participants.

Related Work

Magnien and Vigouroux [6] conducted a study with soft keyboards that uses visual clues such as highlighting keys with a bold font. Twelve subjects were asked to enter lists of words as quickly and accurately as possible. Different keyboard layouts were used such as AZERTY and one similar to METROPOLIS. They found that by highlighting the next likely keys that may be pressed, it increased text-input speed for amateur typists. [6]

Another innovation in soft keyboard features includes dynamic key resizing. Gunawardana et al. [5] developed an algorithm which dynamically resizes the hidden target space of soft keyboard keys as a function of probability. The hidden geometry of the layout is also modified. A predictive component and algorithm determines the both the resizing amount and change in layout geometry, which was shown to increase the accuracy of soft keyboard inputs for users. [5] There are many studies that have explored key target resizing, however they often can be frustrating for users if the resizing is too aggressive. [5] A similar issue could arise with predictive coloring of the keys. A user may be typing a word that the predictive model is unfamiliar with, making it difficult to find the correct next letter that would be darkly colored red.

Al Faraj and Vigouroux. [1] created an apparatus that dynamically increased the visual size of the keys on a soft keyboard. This apparatus used a QWERTY keyboard as the layout for the participants. [1] It was shown that users were more accurate and had higher typing speeds. [1]

Pandey et al. [9] created a system for creating numerical suggestions in the suggestion bar of soft keyboards. Deep learning algorithms are used to predict numerical data using information from text messages [9]. For example, if the content of a text concerns different food prices, price suggestions for food will pop up in a suggestion bar component. [9]. Another example would be the distance from one destination to another [9]. If the conversation is about how far Algonquin Park is, the system will try to accurately predict how far your current location is to Algonquin Park and display that information in the bar above the soft keyboard [9].

Alharbi et al. [2] demonstrated that error correction in mobile communication is difficult to regulate [2]. Predictive keyboards offer solutions to reducing errors, but also negatively affect speed, perception and the quality of spelling [2]. Constant change of user focus between keyboard keys and the suggestion bar is a primary drawback of the suggestion bar component. [2]

Gong et al. [4] designed an improved Multitap text entry system called PNLH. PNLH mainly has two features: next-letter highlighting and next-letter prediction. [4] PNLH makes predictions based on what the user has entered and highlights the three most likely next letters. [4] Experiments prove that PNLH has faster input speed and better accuracy than traditional predictive keyboards. [4] As the number of experiments increases, this difference will be enlarged. [4]

METHOD

Participants

Six volunteer participants performed the experiment, all of which were male. Ages varied, with two-thirds of participants in their 20s, and one-third between 50-70 years old. The majority of participants were experienced soft keyboard users.

Apparatus

The prototype is an android application. It presents a participant with five phrases to input onto a QWERTY soft keyboard. Participant input accuracy is measured using Levenshtein distance. Participants are timed to calculate WPM. There are three different options for the soft keyboard: vanilla, red-green, and green only. The vanilla option has no color cue prediction. The red-green option has negative red coloring for low probabilities, and green coloring for higher probabilities. The green only option leaves lower likelihood keys unmutated, and colors high probability keys green.

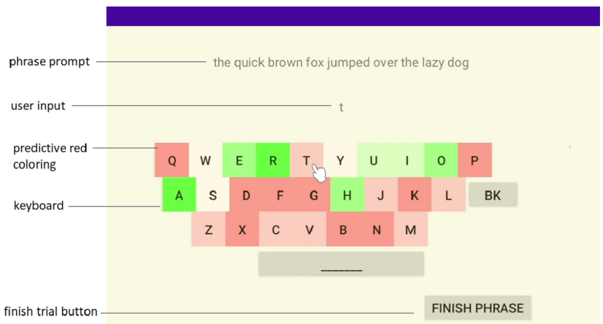

Figure 1. Screenshot of the apparatus soft keyboard with red-green predictive color cues.

For example, a user types the letter “t” during a phrase trial in Figure 1. The prototype uses the substring -- in this case “t” -- as a key in a hashmap. The hashmap retrieves an array of probabilities related to that substring: each cell of the array corresponding to a letter of the English alphabet. With the probabilities available, the prototype uses them to re-color the keys.

The data that has been gathered is easily reproducible. The trials track texting speed and words per minute. The data can be reproduced by simply conducting similar trials on any soft keyboard. The algorithms used to track the entry speed, accuracy, as well as WPM can all be applied to an arbitrary soft keyboard.

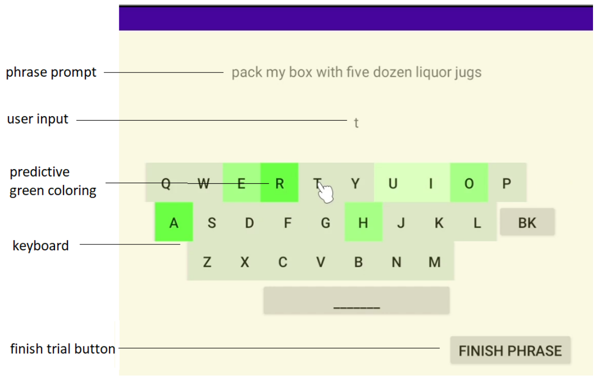

Figure 2. Screenshot the apparatus soft keyboard with green-only predictive color cues.

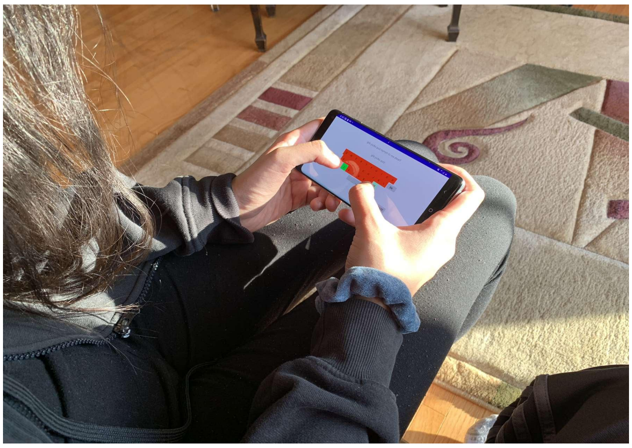

Figure 3. Participant performing input during red-green trial

Procedure

Participants were instructed to complete five trials with each respective option: vanilla, red-green, and green-only. Each trial presented a participant with a new phrase to type. Phrases were identical between each option, and always in the same order between trial blocks. Participants were told that their speed and accuracy per phrase was being measured. Participants were instructed that the timer began at the moment of the first keystroke.

Design

The user study employed a 3 × 5 within-subjects design. There were three independent variables.

- Layout cue option (vanilla, red-green, green-only)
- Trials (1, 2, 3, 4, 5)

The dependent variables were accuracy (using Levenshtein distance), and WPM.

In order to counterbalance the order of testing and offset learning effects, participants were divided into six groups: one participant per group. Each group represented a unique permutation of option orderings. For example, group 1 tried vanilla, red-green, then green-only option last.

Each participant completed 5 trials of each option. The total number of trials was 6 participants × 3 cue options × 5 trials = 90 trials.

The phrases in each trial were pangrams. A pangram is a sentence that contains each letter of the English alphabet at least once. Pangrams were chosen so participants would fully utilize the keyboard layout.

The chosen pangrams contain several uncommon words (quartz, liquor), which may give an advantage towards the layout options with predictive color cues.

RESULTS AND DISCUSSION

Accuracy

Participants were on average 2.02% more accurate with the green-only option over the red-green option. Participants were on average 2.52% more accurate with the green-only option than the vanilla option. These results are graphed in Figure 4.

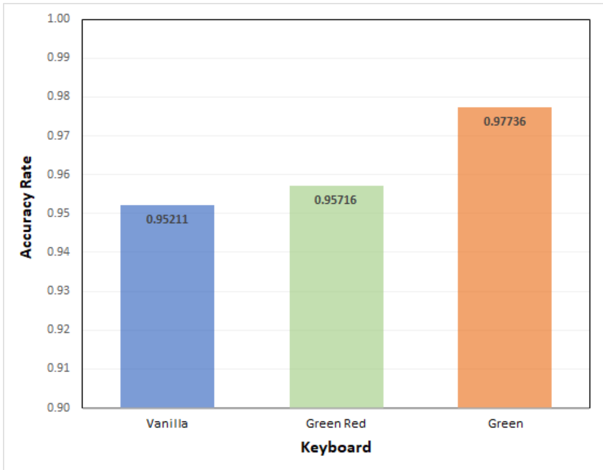

Figure 4. Average accuracy rate for all three keyboard options across all groups

Speed

Participants using the vanilla layout averaged 20.97 WPM, which was 4.55 more WPM than the greed-red option, and 2.0 more WPM than the green-only keyboard. These results are graphed in Figure 5.

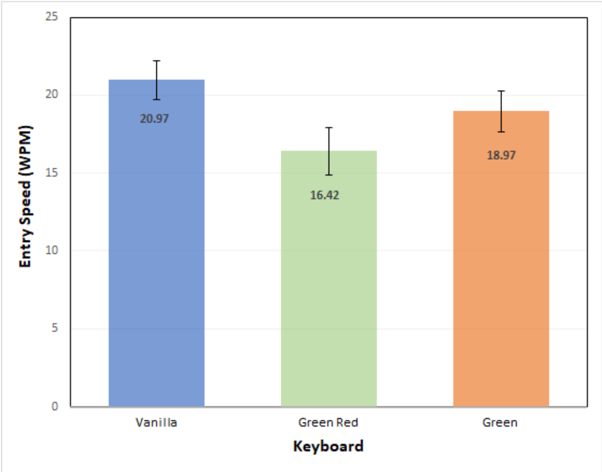

Figure 6. Average WPM for each keyboard option across all groups.

User feedback

An older participant mentioned that he found that the red-green option significantly captured his focus, and started to helpfully lead his attention much more than his memory of the QWERTY layout.

Another participant remarked that the red-green option was moderately more distracting than the green-only option. Distractibility with respect to negative red cues may be causal to the accuracy discrepancy between red-green and green-only options. Another participant preferred the less busy palette of the green only option generally.
Some participants remarked that they were able to type faster using the vanilla option, noting they were less distracted. Distractibility may account for the discrepancy between WPM speeds.

Due to the low number of participants, it is difficult to safely draw any significant conclusions.

CONCLUSION

Participants using the predictive key coloring versus the vanilla option suffered significantly with respect to speed, 15.6% on average. However, they gained a small amount of accuracy, 2.27% on average. The user feedback regarding distractibility is a useful starting point for a further iteration of the prototype. If keys were recolored more subtly, they might prove to be less distracting and increase WPM. Ideally, the same experiment could be repeated with a larger number of novice soft keyboard users. Predictive color cues may have learning applications for soft keyboard use and spelling.

There is room for improvement. The algorithm used to create the hashmap of probabilities was naive. A dictionary of 300,000 English words was scanned, and next-letter occurrences were counted for each candidate substring. One improvement would be to give increased weight of probability to common words. A feedback system for adjusting probability weights could also be added, consuming personal typing habits of users.

Secondly, incorporating the “backspace” and “space” button into the predictive coloring system may help. For example, the backspace button may turn green if a word is misspelled, or the space button may turn a shade of green if a word is complete. However, due to the organic nature of language, it may be difficult to know when a word is being deliberately spelled in an unconventional way.

If the system were to be retooled as a learning apparatus for spelling and text entry, the severity of the color cues could be dynamically drawn back as the user gains proficiency. An interesting aspect of the system is that a user can experiment with new words -- a child or second language-learner may be inclined to follow a path of suggested letters. Once words are completed, it may be useful to display the definition of the word as well.

REFERENCES

1. Al Faraj, M., and Vigouroux, N., A virtual keyboard for mobile devices. *Proceedings of the 13th International Conference on Human-Computer Interaction - MobileHCI 2011*, (New York: ACM, 2011), 3-10
2. Alharbi, O., Stuerzlinger, W., and Putze, F., (2020, November). The Effects of Predictive Features of Mobile Keyboards on Text Entry Speed and Errors. *Proceedings of the ACM on Human Computer Interaction*, (New York: ACM, 2020) 4(183), 1-16.
3. Arnold, K., Chauncey, K., and Gajos, K., Predictive Text Encourages Predictable Writing, *Proceedings of the 2th International Conference on intelligent user interfaces - IUI 2020*, (New York: ACM, 2020), 128-138.
4. Gong, J., Haggerty, B., and Tarasewich, P., An enhanced multitap text entry method with predictive next-letter highlighting. *Extended Abstracts Proceedings of the 2005 Conference on Human Factors in Computing Systems - CHI 2005*, (New York: ACM, 2005), 1399-1402.
5. Gunawardana, A., Paek, T., and Meek, C., Usability guided key-target resizing for soft keyboards. *Proceedings of the 15th International Conference on intelligent user interfaces - IUI 2010,* (New York: ACM, 2010), 111-118.
6. Magnien, L., and Vigouroux, N., Mobile text input with soft keyboards: optimization by means of visual clues. *Proceedings of the 6th International Symposium of Mobile Human-Computer Interaction - MobileHCI 2004*, (New York: ACM, 2004), 197-218.
7. Norman, D., and Fisher, D., Why Alphabetic Keyboards Are Not Easy to Use: Keyboard Layout Doesn’t Much Matter. *Human Factors*, (1984, January), 509-519.
8. Noyes, J., The QWERTY Keyboard: A Review. International Journal of Man-Machine Studies, (1983), 265-281.
9. Pandey, L., Alizadeh, A., and Arif, A. S., Enabling Predictive Number Entry and Editing on Touchscreen - Based Mobile Devices. *Proceedings of the 2020 Conference on Human Information Interaction and Retrieval - CHIIR 2020*, (New York: ACM, 2020), 13-22.
10. Zhai, S., Hunter, M., and Smith, B., The Metropolis Keyboard -- An Exploration of Quantitative Techniques for Virtual Keyboard Design, *Proceedings of the 13th annual symposium on User interface software and technology - UIST 2000*, (New York: ACM, 2000), 119-128.