CentraList

Georges Badr, Mathieu Raynal

CentraList

Georges BADR[a] and Mathieu RAYNAL[a,1]
[a] *IRIT, ELIPSE team*
Université Paul Sabatier
31062 Toulouse cedex 9, France

Abstract. In this paper we propose a novel predictive text entry system with a new distribution of its keys around a word list. Our aim is to optimize the use of software keyboards allied to prediction list. A prediction list is generally used to reduce effort by presenting a list of candidate words that may continue the prefix of the user. However, this technique may tire the user by adding additional cognitive and perceptual efforts. The user has to search continuously for his aimed word in the list, moving his eye-gaze between the list and the keyboard. Our new design allows the user not to lose his focus from the keyboard, no from the list, as well as reducing the movements of the cursor. We also present the experiments carried out to compare our new system to the classical alphabetically ordered keyboard with a list next to the keys.

Keywords. Soft keyboard, interaction, word prediction, prediction list, motor disabled people, eye-gaze

Introduction

Software keyboards were first designed to replace physical or standard keyboards to allow text entry in mobile devices. Text entry is accomplished by clicking or taping on the keys with corresponding letters using fingers or a pointing tool. But researches have shown that the text entry speed is greater when using physical keyboards than when using software keyboards, even for experts. This can be explained by the fact that the user uses ten fingers when typing on a standard keyboard, versus one finger or pointer. On the other hand, people with severe motor disabilities (like myopathy, cerebral palsy, musculoskeletal disorders, etc.), have difficulties using physical keyboards. Generally, they are obliged to use on-screen virtual keyboards. Two main types of interaction are possible: selecting characters (or others items) with a pointing device such as mouse, trackball, eye tracking, etc.; or using an automatic cursor which successively highlights each key. However, in both cases, text input is often slower than with a physical keyboard. Moreover, moving the pointer, repeatedly, over the letters of the keyboard is a tiring job and thus may cause a low text entry speed especially for this kind of people. To remedy this problem, many solutions have been tested since the early 80’s.

[1] Mathieu Raynal, IRIT – ELIPSE team, Université Paul Sabatier, 31062 Toulouse cedex 9, France
Mathieu.raynal@irit.fr
+33 5 61 55 74 08

1. Quick overview on word prediction lists

One technique used to speed up text entry is to present a list of words to the user from which he can select what completes the best his prefix (cf. **Figure 1**).

Figure 1. Keystrokes [4]

As an assistive technology, different types of word lists are presented in the literature. Some of them, such as UKO-II [0], contain a list of fixed word. Some others are displayed next to the last selected character of the keyboard, like PO-Box [2] (cf. **Figure 2** A) or popped up near the writing cursor like in FASTY [3] (cf. **Figure 2** B). Moreover, these lists could be used with pointing devices [2],[4] or with a scanning system [5]. These lists are coupled to a word (or text) prediction system. These systems are also the subject of many researches – in particular in speech processing [6],[7]. The reader may find a survey concerning text prediction systems with word lists in [9].

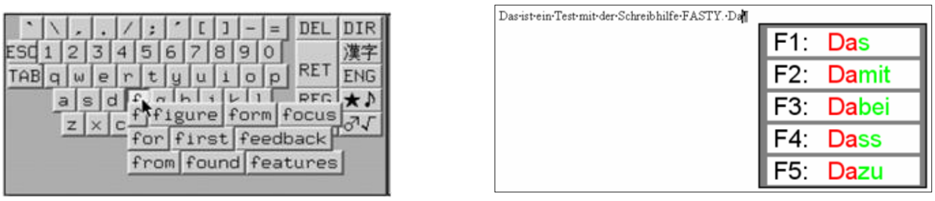

Figure 2. A) PO-BOX interface: The list appears on the selected character, B) FASTY interface: The list appears right next to the cursor.

Adding a prediction list next to the keyboard aimed to minimize the number of keystrokes and reduce the overall distance of the cursor. But experiments have shown that despite the increase in the keystroke saving rate, the text entry rate is still not optimized [10]. This is because of the additional cognitive and perceptual charge caused by the excessive amount of movements of goings and comings between the keyboard and the list. Those movements would make the user lose his focus and eye-gaze from the keyboard and the list. Those causes would dramatically slow the text entry rate and speed, and increase the tiredness of the user.

The present paper focuses on this problem of interaction in the word list: the distance travelled by the cursor and the user's lost of focus. To remedy this shortcoming, we have implemented a new software keyboard (called CentraList), with a new arrangement for its keys and a list of predicted words.

This paper introduces a design for a new text entry system with the hypothesis that it will reduce the cursor movements and the user tiredness. After the description of the principle and of the architecture of this system, we will present the results of the experiments.

2. CentraList

2.1. Principle

Most of word prediction systems studied in the literature displayed the list of propositions next to the keyboard, on the top or below the keys. Other systems like PO-BOX popped up the list above the last selected character. In the first case, the user has to move his cursor or his hand towards the list, and move his eye-gaze away from the keyboard. After searching the word in the list, he has to do the same movements but backwards. In the second case, the list hides an important part of the keyboard, and the fact may annoy the user if he needs to select a character from underneath the list.

Our idea is to conceive a new word prediction interface that aims to minimize the distance between the list and keys. We are proposing a software keyboard, with a new distribution for its keys and a list of prediction. It is an alphabetically ordered keyboard, where the letters are disposed in the form of a rectangle, and the prediction list is displayed in the center. Since the space is the most used character, we conceived our system to have four spaces, one on each side of the rectangle to allow the user to select the nearest space to his cursor (cf. **Figure 3**). By this design, we aim to minimize the distance travelled by the pointer. Our system is named "CentraList" after the fact that the list is in the center of the keyboard.

Figure 3. CentaList interface

The prediction list in CentraList is the same designed and studied by [8]. It is an interactive list, from which the user can click on any character in the word proposed, and the substring to this letter is selected and inserted in the text.

2.2. Architecture

CentraList is divided into three modules: the software keyboard, the prediction system and the presentation of predicted words. Our contribution is on the distribution of the letters in the keyboard. This modular architecture allows to easily adapt our system to any type of prediction algorithm. Its advantage comes from the fact that it is independent of the algorithm: it can be used with any prediction system.

2.3. Prediction algorithm

In this study, CentraList is allied to a prediction system based on a lexicographic tree constructed from a set of words (dictionary). The tree is composed of 364,370 words. Each word is represented by a path from the root to a leaf of the tree, in which each character is a node of the tree (cf. **Figure 4**). Two words with the same prefix use the same initial path to avoid creating too many new nodes. For words which are prefixes for other words, a special leaf "end word" was created (black circle on **Figure 4**). Each arc of the tree is weighted: when a word is added, the system inserts each character in its place by crossing the tree according to what was already entered. The arcs between nodes which are crossed are incremented by 1.

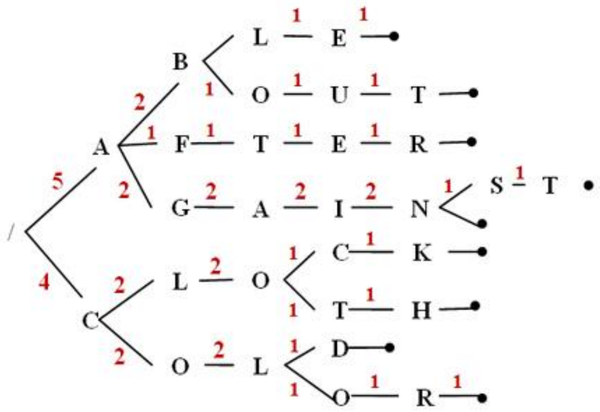

Figure 4. Lexicographical tree with words: able, about, after, again, against, clock, cloth, cold, colour

When the user enters a prefix, the system classifies the characters which can follow according to their probability of appearance (the biggest the value of the arc is, the more the character has chance to appear). This weight will let it sort the predicted words in the list according to their probability defined by their weights.

3. Evaluation

Our hypothesis is that CentraList design would allow reducing the distance to be traveled by the cursor, and that its interactive list would allow reducing the number of keystrokes. Moreover, we believe that, reducing the loss of the focus and minimizing the moves between keys and list, will accelerate input speed and increase text rate.

To validate our hypothesis, a study was undertaken to evaluate CentraList. It was compared to the classical alphabetically ordered layout.

3.1. Participants

There were two types of participants. Eighteen able bodied subjects and seven persons with severe motor disabilities participated in this experiment. They were all volunteers aged between 18 and 35. All participants were regular users of desktop computers and were familiar to pointing device.

3.2. Apparatus

The program was developed with Java SE 6, running on a Toshiba laptop with 2.5 GHz speed and Windows 7 as operating system. Each subject used his own mouse or pointing devices to interact with the soft keyboard. Some disabled people pointed with a joystick.

3.3. Design

Each subject had two exercises of copy (cf. **Figure 5**). The first one was to done on the alphabetic keyboard, and the second on CentraList. Both keyboards comprises only the 26 characters of the Latin alphabet and the space bar. Both systems implemented the interactive prediction list (from which the user can select substrings) and the same prediction algorithm described above.

Able bodied subjects had to copy out 25 phrases containing the most usually used words (total of 984 characters). On the other hand, disabled subject copied 15 sentences consisting of 460 characters. The series of phrases were the same for the two exercises. Subjects were instructed to proceed as quickly and as accurately as possible. We used a counterbalanced, within-subjects design. The independent variable was the design of the keyboard. Dependent variables were text entry speed (Character Per Second - CPS), distance between 2 keystrokes (KeyStroke Per Character - KSPC) and error rate.

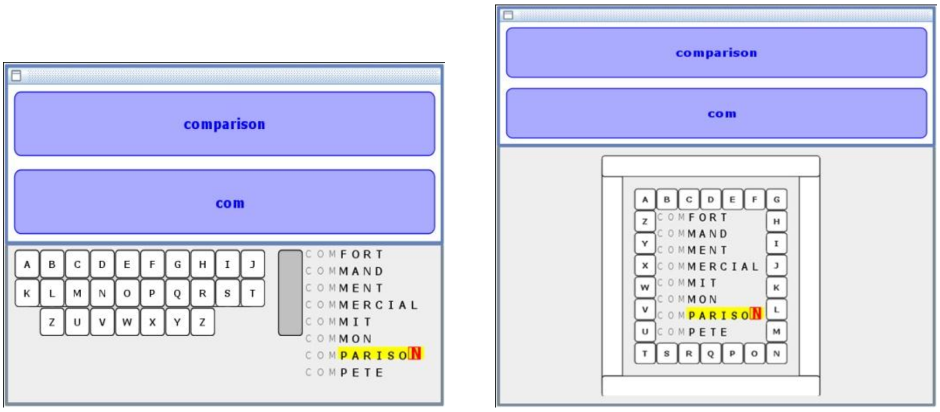

Figure 5. Experiment screen with both classic keyboard (left) and CentraList (right)

3.4. Procedure

The word to be copied was presented on a line, and the word being typed by the user appeared on the line below (cf. **Figure 5**). The text entry errors were not displayed on the screen. Instead there was a visual and audio feedback signaling the error and the strip did not move until the subject entered the right character. At the end of each word, the participant had to hit the space bar.

4. Results

4.1. Text entry speed

The analysis of the results (cf.

Table 1) shows that typing is faster with CentraList than using the alphabetic keyboard. In fact, we can see an increment in the number of characters clicked in one second. On average, CPS measure, for able bodied subjects, is equal to 0.81 with alphabetic keyboard, whereas it is 0.9 with CentraList. Thus the user can gain in average 11.11 % of time with CentraList compared to the other keyboard. On the other hand, the average of CPS for disabled users increased from 0.46 to 0.52, scoring a profit of 13.04 % of time.

Table 1. Performances of different people with the different techniques

		Able bodied people	**Disabled people**
Alphabetic keyboard			
	KSPC	0.84	0.8
	CPS	0.81	0.46
CentraList			
	KSPC	0.82	0.77
	CPS	0.9	0.52

Moreover,

Table 1 shows a reduction of the number of operations needed to be done with CentraList compared with the other keyboard. On average, healthy users needed 0.82 clicks per character with CentraList, slightly less than their need of 0.84 clicks to enter the same character with the old design. Thus they gain an average 2.4%. Secondly, the average KSPC for handicapped subjects decreased from 0.8 to 0.77, reducing the number of operations by 3.9%.

4.2. Error Rate

During the exercises, when the current character differed from the expected character, the error was recorded. The average of error rate for able bodied people was 1.9% for the classical keyboard and 1.79% with the Centralist system.
As for disabled people, the average of error rate with the classic keyboard is 2.58% and 2.45% when using CentraList.

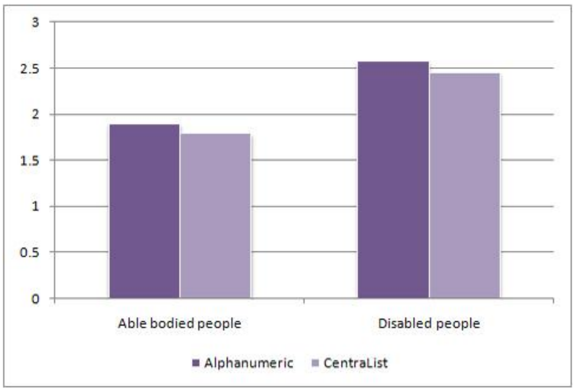

Figure 6. Error rates for both kind of subjects by input method

4.3. Distance between two strokes

Our main goal from conceiving CentraList with this design was to minimize the distance between keys and the list, and reduce the amount of cursor movements so that the user won't lose his eye-gaze. The analysis of the results of the experiments shows that our hypothesis was confirmed. Thus, we can observe a reduction in term of distance the cursor has to do between two clicks. This decrease is shown for able bodied subjects as well as for disabled users. In fact, the average distance between 2 operations in the classical keyboard was 336.78 pixels, and decreases to 304.40 pixels for CentraList. This will give the user a benefit of about 10%. This gain increases to 23.24% of distance economy for disabled people, as the distance decreases from 460.54 with the alphabetic keyboard to 353.51 pixels with CentraList.

Note that users were familiar to the QWERTY keyboard and not the alphabetic one. Thus, both tested systems have new keys distributions for them. We believe that with more experience and more exercises, our results would be better, as users would increase their experience.

5. Conclusion

In this article, we presented a new design for a software keyboard with an interactive list of prediction. The letters are distributed in alphabetic order, with the shape of a rectangle around the list. The aim of this design was to reduce the movement one has to do with his cursor between keys and the list.

We also presented the results of the evaluation conducted with 18 healthy people and 7 disabled persons. The results reported by our experiments show that the CentraList system can be beneficial and advantageous for text entry compared to the classical alphabetic keyboard. In fact, we have obtained a text entry speed augmentation and a reduction of the number of operations necessary for entering a word as well as an important decrease in distance between 2 clicks. Thus, with CentraList, a given word could be entered not only with a reduced number of interactions, but also with less moving between the keys and the list. This would reduce the fatigue and the cognitive charge for both disabled and non-disabled people.

References

[1] Harbusch, K., Hasan, S., Hoffmann, H., Kühn, M., and Schüler, B. 2003. Domain-specific disambiguation for typing with ambiguous keyboards. In Proceedings of the 2003 EACL Workshop on Language Modeling For Text Entry Methods. ACL Workshops. Association for Computational Linguistics, Morristown, NJ, 67-74, 2003.

[2] Masui T., An efficient text input method for pen-based computers, Proceedings of the ACM Conference on Human Factors in Computing Systems CHI'98, 1998.

[3] Beck C., Seisenbacher G., Edlemayer G., and Zagler W.L., First User Test Results with the Predictive Typing System FASTY

[4] Keystrokes on-screen keyboard, user manual, http://www.assistiveware.com/files/KeyStrokesXmanualA4.pdf.

[5] Wandmacher T., Antoine J.-Y., Poirier F., Departe J.-P. SIBYLLE, An Assistive Communication System Adapting to the Context and Its User. ACM Transactions on Accessible Computing, vol.1, issue 1, p. 1-30, 2008

[6] Aliprandi C., Carmignani N., and Mancarella P., An inflected-Sensitive Letter and Word Prediction System, International Journal of Computing & Information Sciences Vol. 5, No. 2, on-Line, p79-85, 2007

[7] Hunnicutt S., Nozadze L., Chikoidze G., Russian Word Prediction with Morphological Support,. 5th International Symposium on Language, Logic and. Computation, Tbilisi, Georgia, 2003

[8] Badr G. and Raynal M., WordTree: Results of a Word Prediction System Presented thanks to a Tree., in International Conference on Universal Access in Human-Computer Interaction, Vol. 5616/2009, Springer-Verlag, p. 463-471, San Diego (USA), 2009

[9] Garay-Vitoria, N., Abascal, J. Text Prediction Systems: a survey. Univ Access Information Society vol 4. Springer-Verlag (2006), 188–203

[10] Newell AF, Booth L, Beattie W. Predictive text entry with PAL and children with learning difficulties. *Britich Journal of Educaction Technology,* 1991; 22(1), 23-40.