



HAL
open science

WordGlass: additional keys to present the most likely words

Mathieu Raynal

► **To cite this version:**

Mathieu Raynal. WordGlass: additional keys to present the most likely words. International Conference on Computers Helping People with Special Needs, Jul 2024, Linz, Austria. hal-04947088

HAL Id: hal-04947088

<https://hal.science/hal-04947088v1>

Submitted on 14 Feb 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

WordGlass: Additional Keys to Present the Most Likely Words

Mathieu Raynal¹

ELIPSE team, IRIT lab, University of Toulouse, France
mathieu.raynal@irit.fr

Abstract. In this article, we propose WordGlass, a system of additional keys that are dynamically added as the user types. Each key proposes one of the most likely words. The keys have been spaced out on the keyboard so that one additional key can be displayed above and another below the last key pressed, without obscuring the keys already present. First results show that, WordGlass has some benefits: the prediction use rate is much higher with WordGlass than with WordList. The majority of participants said that it was easier to see a word that was close to the pointer than to look beside the keyboard. This proximity to the cursor was also reflected in the distance covered by the pointer. On average, participants covered 48% less distance with WordGlass than with WordList. With this reduction of the travelled distance and the better use of the words proposed, the participants were 5% faster with WordGlass than with WordList.

Keywords: motor disability · text entry · assistive technologies · word prediction

1 Introduction

People with motor impairments have difficulty accessing text input devices. To address these needs, a range of assistive technologies have been developed in the form of virtual keyboards [14]. However, numerous studies show that the speed of text input using a soft keyboard remains relatively slow [4,7,10]. In response to this observation, some keyboards are enhanced with linguistic prediction modules attempting to make text input faster. On the one hand, the use of character prediction makes it possible to dynamically reorganize the key layout in order to optimize access to the most likely characters given the text already entered [1,9,15]. The main limitation of this solution lies in the constant reorganization of the keyboard, which can disturb the user. In fact, this dynamic reorganization is limited to scanning keyboard [16].

Another solution being considered to accelerate typing is the use of word prediction, which makes it possible to limit the number of characters to be typed to compose a message. Word prediction models propose a list of the most likely words to be typed (for example, UKO [5], Centralist [3] or POBox [8] presented in Fig. 1). The words displayed are suggested according to the words already

entered and the first characters of the word currently being entered. Word prediction has also proved its effectiveness, reducing the number of characters to be typed by two. However, its use has a strong impact on the cognitive load of the user, who has to observe both the virtual keyboard and the word prediction list. The few experimental studies carried out on the subject have shown that the improvement in typing speed is limited, and that the use of effective prediction is much lower than its theoretical optimum use [16,13].

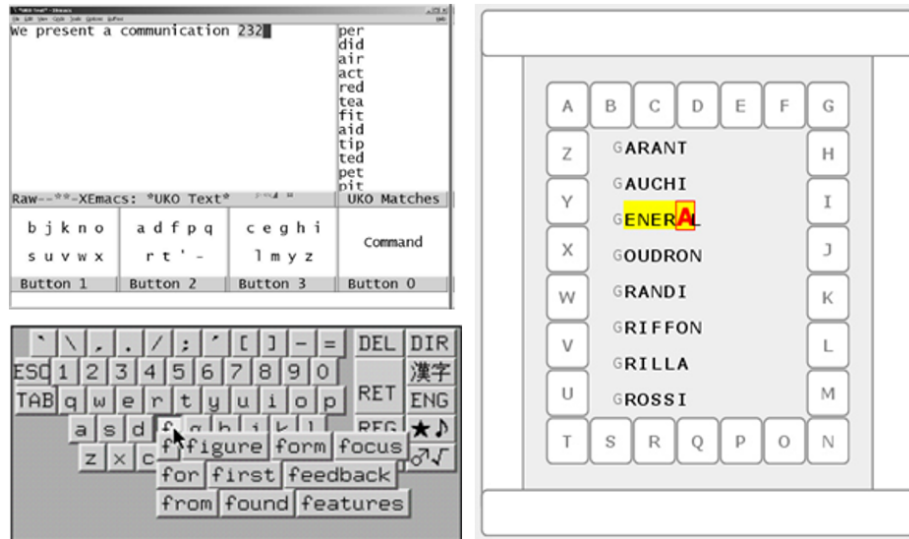


Fig. 1: At the top left, the UKO keyboard with a list on the right of the keyboard; at the bottom left, PoBox with the list of words near the pointer on the keyboard; at right, Centralist with the characters arranged around the list.

2 WordGlass

In this article, we propose WordGlass, a system of additional keys that are dynamically added as the user types. Each key proposes one of the most likely words. The POBox system also offered a list of words near the cursor. The advantage of this technique is that the user can focus on the keyboard without having to look at something nearby. Moreover, it also limits cursor movements when the user wants to select a word from the list. But, the disadvantage of POBox is that the word list covers a part of the keyboard and makes it difficult to access the characters underneath. To remedy this problem, WordGlass uses the same principle as KeyGlass [12], where characters are suggested on additional keys. These keys appear around the last character entered and suggest the 4 most likely characters (Fig. 2). These keys are arranged so as not to obscure the other

keys on the keyboard. To achieve this, the keys on the keyboard are hexagonal, leaving a space at each corner.

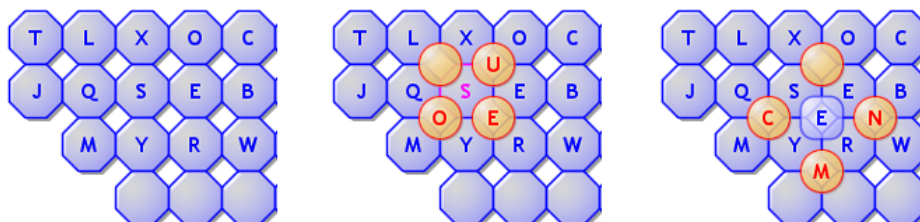


Fig. 2: At left, part of the keyboard layout ; at middle: KeyGlasses suggested after entering the 'S' ; at right : KeyGlasses that appear after entering the 'E' that has been placed on a KeyGlass.

WordGlass adopts the principle of placing additional keys with the most likely words close to the last character typed. The keys have been spaced out on the keyboard so that one key can be displayed above and another below the last key pressed, without obscuring the keys already present (Fig. 3). As users only look at the 2 or 3 words in the list [13], we have also chosen to keep only the 2 most likely words.

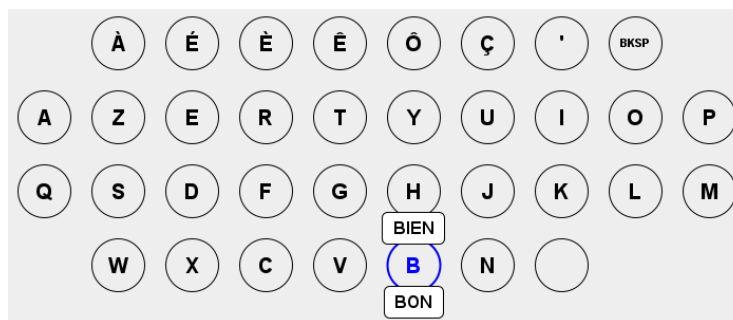


Fig. 3: Additional keys available after entering the 'B'

Our hypotheses are as follows: **(H1)**, the user will have the proposed words in his field of vision and will therefore be more likely to see the proposed words. As a result, he will use them more than those proposed in a list displayed close to the keyboard. **(H2)**, as the words are closer to the pointer, the distance travelled by the pointer will be reduced with WordGlass.

3 Method

To test WordGlass, we conducted a first experiment comparing WordGlass with a traditional word list from the right of the keyboard.

3.1 Participants

Ten participants, three females and seven males, from 18 to 27 years old, took part in the experiment.

3.2 Apparatus

The experiment was conducted on a laptop with a resolution of 1920×1080 pixels. Participants interacted with the soft keyboard through a mouse. Keyboard layout is an AZERTY which was restricted to the 26 characters of the Latin alphabet, the backspace key, the space bar and the seven accented characters most commonly used in the participants' native language. The soft keyboard was developed in Java SE and its Swing library. The soft keyboard with the traditional word list, called WordList in this article, offers a list of four words on the right of the keyboard (Fig. 4).

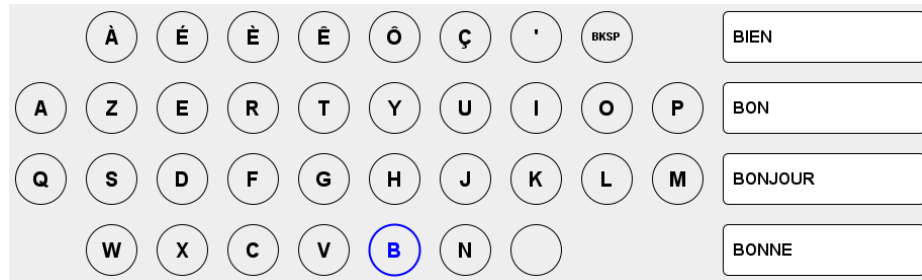


Fig. 4: WordList with the four words after entering the ‘B’

3.3 Procedure

The task to be performed is a copy task. The sentence to be copied was presented on a line, and the sentence being typed by the user appeared on the line below. Participants were asked to enter as many sentences as possible per exercise, and as accurately as possible. Each exercise lasted 10 minutes. Once the 10 minutes had elapsed, the exercise stopped when the participant validated the sentence being typed. The sentences are randomly selected from a corpus of 90 sentences [6]. Each sentence contains words chosen among the most common ones and are statistically representative of the participant's language. A learning phase, consisting of entering a randomly selected sentence, preceded each exercise.

3.4 Design

The experiment was a within-subjects design with word prediction method (WordGlass, WordList) as independent variable. The dependent variables were text entry speed (cps), prediction use rate (%), and distance (pixel).

Participants were divided into two groups for counterbalancing and were randomly assigned to two groups of four. One group started with WordGlass and the other started with the WordList.

4 Results

We present below the results of our experiment, organized by dependent variable.

4.1 Text entry speed

We computed the text entry speed per sentence. This was calculated by dividing the length of the sentence (including space between words and at the end of sentence) by the time (in seconds) to enter this sentence. Thus, this text entry speed is done in characters per second (cps). Participants were on average 5% faster with WordGlass than with WordList (1,368 cps and 1,302 cps respectively), but this difference was not significant ($F_{1,9} = 1,561$, ns).

4.2 Prediction use rate

The prediction rate is the ratio between the number of times the word to be entered is proposed to the user and the number of characters entered on the keyboard. The prediction rate was 44.41% with WordList compared with 33.88% for WordGlass. This difference is explained by the fact that WordList has four words, whereas WordGlass has only two. We also calculated the prediction use rate: i.e. the number of times the participants selected the word compared to the number of times the word was proposed. Participants used the words proposed on WordGlass in 79.29% of cases, compared with only 58.14% when the words were presented on WordList. The prediction use rate effect was statistically significant ($F_{1,9} = 15.234$, $p < .005$).

4.3 Distance

Finally, we calculated the mean distance travelled by the pointer to enter a character. The mean distance is the total distance travelled by the pointer to select the different elements (characters or words) divided by the total number of characters entered. For words where the participants selected one of the proposed words, the average distance covered by the pointer was 109 pixels with WordGlass and 210 pixels with WordList. This represents a 48% reduction in the distance travelled by the pointer.

5 Discussion and future work

This first study confirms our hypotheses: participants used the words suggested in WordGlass more often than in WordList (**H1**). The majority of participants also said that it is easier to see a word close to the pointer than to look beside the keyboard. Similarly, the calculation of the travelled distance shows that the pointer travelled almost 50% less distance with WordGlass when the participants were using the suggested words (**H2**). In addition, when there were few characters left to type on the current word, the participants recognized that they preferred to continue typing on the keyboard rather than moving to the list.

With this reduction of the travelled distance and the better use of the words proposed, the participants were 5% faster with WordGlass than with WordList, but this difference is not significant. On the other hand, the text input speed is not very high compared with text input speed on physical devices. But it is in the range of text input speed observed on soft keyboards [11] and it is higher than on comparable systems [2].

Finally, some participants told us that the advantage of WordList was that it offered more words and that, as a result, the word they were looking for could appear earlier in the suggestions. We will carry out an additional study with four words proposed for WordGlass. two words will be proposed above the last character entered and two others will be proposed below. This additional study will enable us to validate the best choice between two or four words proposed around the key, and to study whether WordGlass with four words increases the user's typing performance or degrades it compared with WordGlass with two words.

Acknowledgements

This work was funded by the French National Research Agency (ANR) as part of the AAC4All project (ANR-21-CE19-0051)¹

References

1. Aulagner, G., François, R., Martin, B., Michel, D., Raynal, M.: Floodkey: increasing software keyboard keys by reducing needless ones without occultation. In: Proceedings of the 10th WSEAS international conference on Applied computer science. pp. 412–417. World Scientific and Engineering Academy and Society (WSEAS) (2010)
2. Badr, G., Raynal, M.: Wordtree: results of a word prediction system presented thanks to a tree. In: 5th International Conference on Universal Access in Human-Computer Interaction. Applications and Services, UAHCI 2009. pp. 463–471. Springer (2009)
3. Badr, G., Raynal, M.: Centralist. In: Everyday Technology for Independence and Care, pp. 944–951. IOS Press (2011)

¹ <https://www.aac4all.org>

4. Felzer, T., MacKenzie, I.S., Beckerle, P., Rinderknecht, S.: Qanti: a software tool for quick ambiguous non-standard text input. In: 12th International Conference on Computers Helping People with Special Needs, ICCHP'10. pp. 128–135. Springer (2010)
5. Harbusch, K., Hasan, S., Hoffmann, H., Kühn, M., Schüler, B.: Domain-specific disambiguation for typing with ambiguous keyboards. In: Proceedings of the 2003 EACL Workshop on Language Modeling for Text Entry Methods (2003)
6. Hoiry, L., Khelil, C.B., Rayar, F., Halftermeyer, A., Antoine, J.Y., Raynal, M.: Corpus de phrases pour l'évaluation de systèmes de saisie de texte en français. Tech. rep. (2024)
7. Jabeen, F., Tao, L.: An efficient text entry model for scanning ambiguous keyboard. In: 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC). vol. 1, pp. 71–76 (2017)
8. Masui, T.: Pobox: An efficient text input method for handheld and ubiquitous computers. In: International Symposium on Handheld and Ubiquitous Computing. pp. 289–300. Springer (1999)
9. Merlin, B., Raynal, M.: Evaluation of spreadkey system with motor impaired users. In: International Conference on Computers for Handicapped Persons. pp. 112–119. Springer (2010)
10. Poláček, O., Mikovec, Z., Slavik, P.: Predictive scanning keyboard operated by hissing. In: Proceedings of the 2nd IASTED International Conference Assistive Technologies. pp. 862–9. Citeseer (2012)
11. Polacek, O., Sporka, A.J., Slavik, P.: Text input for motor-impaired people. *Universal Access in the Information Society* **16**, 51–72 (2017)
12. Raynal, M.: Keyglasses: semi-transparent keys on soft keyboard. In: Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility. pp. 347–349 (2014)
13. Raynal, M., Badr, G.: Study of user behavior when using a list of predicted words. In: Joint International Conference on Digital Inclusion, Assistive Technology & Accessibility - ICCHP-AAATE 2022. pp. 331–337 (2022)
14. Raynal, M., Badr, G., MacKenzie, I.S.: Dessk: description space for soft keyboards. In: International Conference on Human-Computer Interaction. pp. 109–125. Springer (2022)
15. Raynal, M., MacKenzie, I.S., Merlin, B.: Semantic keyboard: Fast movements between keys of a soft keyboard. In: 14th International Conference on Computers Helping People with Special Needs, ICCHP'14. pp. 195–202. Springer (2014)
16. Wandmacher, T., Antoine, J.Y., Poirier, F., Départe, J.P.: Sibylle, an assistive communication system adapting to the context and its user. *ACM Transactions on Accessible Computing (TACCESS)* **1**(1), 1–30 (2008)