Novel Key Input Technique

Our novel key input technique uses auto-completion to enable faster typing. After a user has entered at least two characters, s/he is offered a list of candidate words based on his/her last letters. The words used for auto-completion are taken from the text. By using the arrow keys and Enter, s/he can select the matching word. The user can also manually invoke auto-completion by pressing the Shift and Space keys. This key combination was chosen because it is easy to grasp. The words for auto-completion are sorted alphabetically. Our auto-completion is based on the Custom Completer from the official Qt Documentation[1] and small code adjustments were made. To measure whether auto-complete saves time, the typed keys were logged with timestamps from which the words per minute (WPM) were calculated. The calculation depends on the total amount of time elapsed and the total number of characters in the text. It is also noted if a word or sentence is complete with the last entered key and if the task is finished. A sentence ends when a newline is entered and a word ends when a punctuation or whitespace was typed. The participants id, the name of the text file and the last typed sentence on screen is shown on stdout. It is also logged whether the normal keyboard was used or with auto-completion, the numbers of characters when auto-completion should appear and which key was pressed (code and text value). The time how long a user needs to type a word and sentence is printed in seconds.

Limitations

Auto-completion requires at least two keystrokes and will therefore only gain advantages with long words, whereas it will be slower with short words. An advantage of the method is to avoid typos by using predefined words. For the study, a static limited word list was used because it requires little memory and allows the time needed to display the alternative words to be kept low. The list can be customised, but is not suitable for larger general text tasks (in our example).

Experimental Setup

In order to evaluate the new key input technology, we conducted a small study. We want to compare how the auto-completion influences the typing speed calculated in WPM and the time the user needs to type one sentence.

Apparatus

The study was performed at home with a Tuxedo notebook and an external Cherry keyboard with Qwerty design. We installed and started a Python script which displays the texts and offers a text input area. It was decided to set a text copying task, which usually has two "focus of attention (FOA)" (see MacKenzie & Soukor-eff (2002)). In contrast to a text creation task, here the text is copied and does not have to be reproduced from memory. German texts were selected based on the nationality of the participants. These were displayed on the screen and below the participants could see what they had entered.

1 see https://doc.qt.io/qt-5/qtwidgets-tools-customcompleter-example.html

Design

The study was conceived as within-subject design, implying that the participants typed texts with and without auto-completion.

The hypothesis is that, on average, you can write more WPM with auto-completion than without and that you are faster in typing an sentence when you use auto-complete. This results in the measured WPM and the measured time in seconds the user needs to finish a sentence as the dependent variables. The independent variables are the texts to be typed and the task (with or without auto-completion).

As control variables we have the design of the keyboard as a text input device, the window size (810 x 600 pixels) and the global position of the window in the centre of the screen.

Procedure

First, the participant was asked about his/her age, occupation and his/her gender was noted. The procedure of the study was explained to him/her and any questions were clarified. To minimise learning effects and repetition, participants had to copy two different German texts. They were asked to type each text once with and once without auto-completion. The conditions and texts were set via config files. These were accessed one after the other in a counter-balanced order[2], as shown in the table 1. The participant has finished the study as soon as he/she has completed four rounds with the corresponding conditions.

id	task 1	task 2	task 3	task 4
1	normal typing text of aurel	normal typing text mastrocola	auto-complete text of mastrocola	auto-complete text of aurel
2	normal typing text mastrocola	auto-complete text of aurel	normal typing text of aurel	auto-complete text of mastrocola
3	auto-complete text of aurel	auto-complete text of mastrocola	normal typing text mastrocola	normal typing text of aurel
4	auto-complete text of mastrocola	normal typing text of aurel	auto-complete text of aurel	normal typing text mastrocola

Table 1: Order of conditions

We conducted a pilot study to test the application and the process of the study. Based on the results of the preliminary study, we removed punctuation marks from the auto-complete word list as they were confusing and unnecessary. We have also decided to sort the list alphabetically for faster access. During the pilot study, we noticed that when logging our data, the first values regarding WPM do not make sense, because the time measurement first starts when a character is entered. Due to lack of time, this has not been changed, as the user can read through the entire text once and then start typing. Thus, we have only included the last WPM in our results.

2 source: https://cs.uwaterloo.ca/\~dmasson/tools/latin_square/

Participants

Due to the current pandemic situation the selection of participants is limited. We had four participants (one male and three female) with the age of 20, 22, 25 and 38. The participants were two media informatics master students, one software developer and one student of international and cultural business study.

Results

In terms of both the average time taken to type a sentence and the average number of WPM, the two input methods are close to each other. The average time for typing a sentence is 10.08 seconds (SD = 5.72) for normal typing and 11.06 seconds (SD = 6.69) for auto-complete.

The average number of WPM is 0.88 (SD = 0.77) for normal typing and 0.97 (SD = 0.72) for auto-complete. The boxplots 1 and 2 visualise the results.

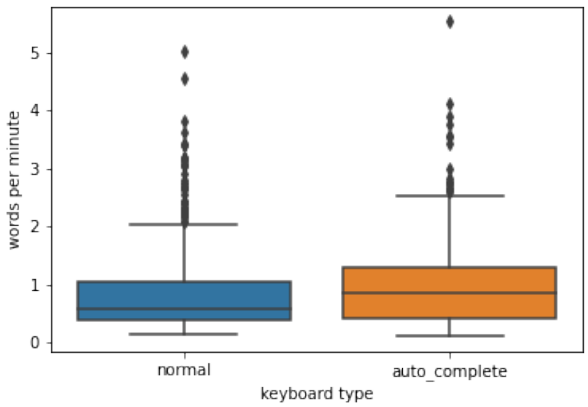


Figure 1: Boxplot with words per minutes

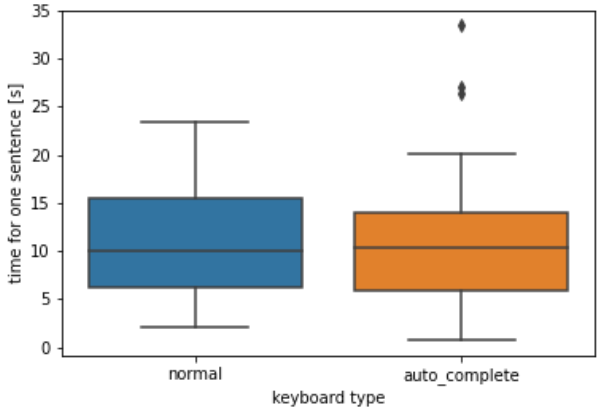
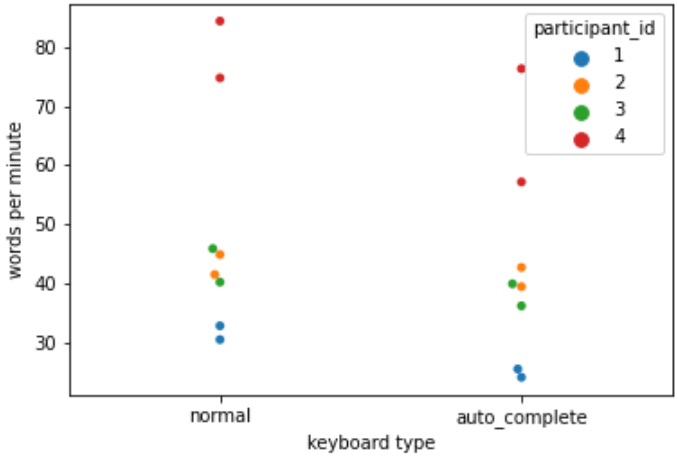


Figure 2: Boxplot with time for one sentence

The swarm plot shows that the measured number of typed words per minute is very different from participant to participant and that the data is highly scattered.



The collected data (WPM, time to complete a sentence in seconds) were tested for variance homogeneity and normal distribution, as these are the conditions for a t-test (Field,2009). Since none of the above is true, the Welch-Anova-Test was chosen to test the input methods for significant differences.

The Welch-Anova test shows that the differences regarding the number of words per minute between normal typing and auto-complete are not significant ($t(1) = 0.33$, $p = 0.57$). The differences among the two typing techniques regarding the time it takes to type a sentence are also not significant ($t(1) = 0.83$, $p = 0.04$).

This indicates that the hypothesis is not confirmed. Auto-completion does not reduce the time required to type a text.

Discussion

We only had a limited amount of participants so the results must be viewed critically. Moreover, skilled writers consider only the text and not the keyboard, hence they only have one FOA whereas unskilled ones have two FOA (MacKenzie&Souko-reff,2002). With auto-completion, the participants have to switch from the text to the selection list and still make a selection, so FOA increases and the typing speed decreases. In addition, their back-and-forth switching interrupts the writing flow, so that auto-completion has a disadvantage for them in terms of typing speed.

Another aspect is that they first have to familiarise themselves with the tool, which also takes time. A further study would need to examine how learning the tool affects speed by testing over a longer period of time.

Nevertheless, it has been observed that auto-complete reduces typing errors, but this observation would need to be verified in a further experiment.

References

Field, A. P. (2009).Discovering statistics using SPSS: and sex, drugs and rock ’n’ roll (3rd ed ed.). Los Angeles: SAGE Publications.

MacKenzie, I. S., & Soukoreff, R. W. (2002). Text entry for mobile computing:Models and methods, theory and practice.Human–Computer Interaction,17(2-3),147–198.