Introduction

The goal of this paper is to give an overview of our implemented input method, as well as its evaluation through a user study. In the user study, we compare text input with an unenhanced, traditional keyboard and text input with an autocomplete enabled, traditional keyboard. Since the assignment doesn't need us to track typing errors, we simply compare the time it takes to enter a self-created set of words and sentences with both input methods. In the following text we describe the implemented input method, as well as the test setup, our hypotheses and our findings.

Autocomplete as Input Method

We decided to implement and evaluate the autocompletion of certain words as the input method of our choice. Once the user starts typing the letters of a word, autocompletion hints are shown to him, and he may select one of them as the accepted word.

Design Decisions

When we decided to use autocompletion as our input method, we first looked at common autocompletion paradigms and thought about the most important aspects for our use case. We decided that the display and the selection of the autocompletion hints was the most important part to consider there. While in mobile context it is custom to only display a few suggestions and place them next to each other, we decided that for desktop computers or laptops with much more display size, we would rather display all the suggested words. The selection can then be done using the up and down arrow keys, which are not directly needed to enter any from of text, especially in our input paradigm with a one-lined input field that takes no line breaks.

Thus, we decided to use popup completion, where all the possible hints are displayed in a popup under the entered text. The user is able to directly choose the desired word by navigating the popup list using the arrow-down and arrow-up keys and can delete characters from the currently displayed word to get a broader amount of autocompleted suggestions. Only words that completely match the currently typed word from its start to finish are propsed as autocompletion entries. When the user uses the space key to start a new word, the autocompletion mechanism is reset to give proposed outcomes for the new word.

Technical Implementation

For the implementation of our input method, we used PyQt's QLineEdit for entering text within a line and combined it with the QCompleter class and a custom wordlist for the completers QStringListModel. We use the QCompleter's PopupCompletion to display the autocorrection hints in a popup list and split the user's entered text at every space, so that he can get new autocomplete suggestions whenever he starts typing in a new word.

Limitations

Since our input method is mostly for the course and thus rather basically implemented and not refined, it still has to overcome a bunch of limitations before it could be commonly used. One limitation is that the QCompleter's PopupCompletion uses the arrow keys without any key modifier like shift to navigate the hints. This makes it better usable if you only type in one line since you don't need to press too many keys, but at the same time it is suboptimal if used in a scenario where the user wishes to navigate between multiple lines using the up and down arrow keys. The second limitation is that the autocompletion hints currently come from a wordlist we gave the completer at the start. That means it only knows suggestions we gave it beforehand and will never complete any other words. For a more common use it could be extended to add words that the user typed to the list if those weren't already present.

Evaluation

To compare the speed with which users are able to enter text with our input method and an unenhanced text input, we conducted a small user study. In accordance with the assignment, we tracked the time needed to enter the text, but did ignore typing errors for both input methods.

We expected that a user's text input speed should differ when using the autocompletion method compared to the classical approach, but were not sure if autocompletion would make them faster (due to working as intended and decreasing the amount of keypresses needed for long words) or slower (due to the time it took to select the appropriate autocorrected suggestion). Thus, we tested the hypotheses:

H0: The time needed to type a sentence with the autocompletion method does not differ from the time needed to type a sentence with normal text input.

H1: The time needed to type a sentence with the autocompletion method differs from the time needed to type a sentence with normal text input.

Experimental Setup

We used a rather simple experimental design to compare the text input speed of both input methods. Since we only one factor in the used input technique, we also have only two different trials in the input with autocompletion and the one without.

Execution

At the beginning of the study, users received the instructions that they would be shown a text on their screen, and that they should enter this text in the textbox displayed below it. Once they were finished with the given sentence, they were to confirm it through pressing the enter key. Afterwards the next sentence would be displayed to them, and so forth until they reached the end of the experiment. They also received a demonstration of how the autocompletion and the choice of the hint works, so that they would not skew the time taken for the first takes caused by them having to familiarize with the navigation of the hints.

Users input the displayed text, while every input key as well as every finished word or sentence is logged in the csv-format, with a timestamp and the time that it took to enter it, as well as the condition that it belongs to and other relevant information.

Four participants were tested in total and each of them was shown ten sentences. The first two participants received their ten sentences with every second one of them with the normal input method, the others with autocompletion. It was the same for the next two participants, only the order of the sentences was reversed this time. Thus, every sentence was done twice with and twice without autocompletion, as well as rather at the start and at the end of the experiment.

Participants

The experiment was conducted with four participants. The participants had all experience with computers, were in the same age and had no health or wrist issues effecting their typing speed. We asked possible participants about their typing skill before the experiment was conducted and chose four participants who all assessed their typing skills to roughly be on the desired level. The desired level of typing skill was that participants shouldn't be completely fluid in ten finger typing, but should regularly work with computers and have a decent typing speed.

Variables

- The independent variable in our experiment is the type of input method used (Normal text input vs text input with autocompletion)

- The dependent variable is the time that it takes a user to write a sentence

- Controlled variables in that experiment were the used keyboard, and the amount of typing skill that the participants possessed. All participants used the same keyboard and computer to conduct the experiment. All participants were selected to have a similar and decently good level of typing skills.

- We tried to limit our confounding variables. Therefore, we conducted the experiment in a laboratory where no other people and thus no confounding external distractions and interruptions were present. We changed the order of sentences so that no sentence would only be done by eager or too exhausted participants, and had every sentence done as often with autocomplete as without it.

- Random variables such as a user's mental abilities were hard to quantify and prevent, but since all participants visited an university, it can be assumed that they shouldn't differ too strong in their mental capabilities.

Results

The following boxplots show the captured time in milliseconds per sentence, displayed after participants. Participant 2 had the median with the biggest input time for his sentences with around 13 seconds.

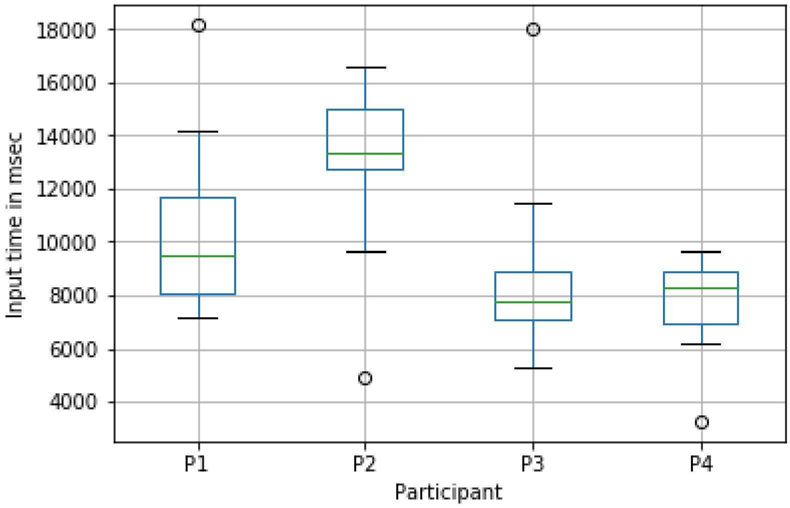

Figure 1: Input time per sentence for the participants

To evaluate our initial hypothesis, we grouped the input time for each sentence in our captured dataset by the two researched conditions (IT: input with autocomplete technique, NT: normal input technique) and then conducted a t-test to see whether or not they significantly differed from one another. The boxplots for the sentence lengths of the two conditions are shown below.

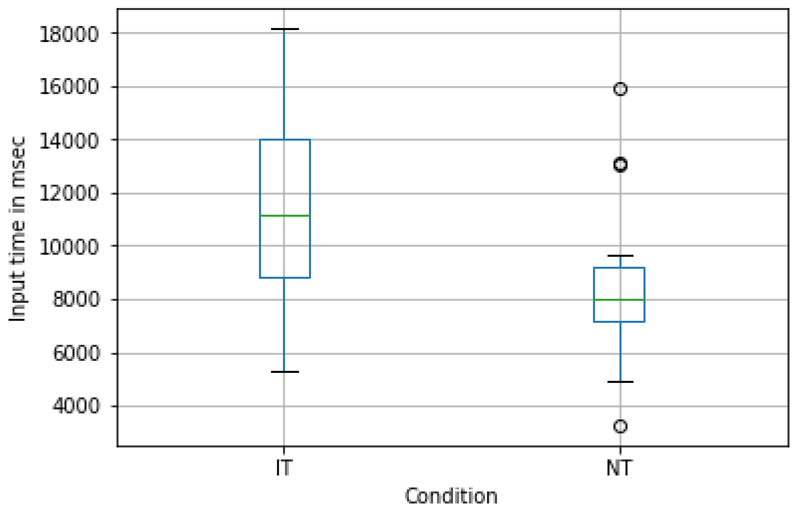

Figure 2: Input time per sentence for IT and NT

The conducted t-test produced a p-value of around 0.005, which means there is a statistically significant difference between the text input times with autocomplete and in the normal way. As seen in figure 2, the input times for the autocompletion were bigger than those for the normal input technique. We thus concluded that our autocompletion implementation did not help in enhancing the user's typing speed. It did the exact opposite and slowed him down, probably because in our implementation the user needs to mentally process and select the autocompletion hint, which took longer than just typing the word in the normal way. In addition, we also tested with relatively short words and sentences, whereas the autocompletion function might be more useful or speed enhancing with longer words.