# How We Swipe: A Large-scale Shape-writing Dataset and Empirical Findings

LUIS A. LEIVA[*], University of Luxembourg, Luxembourg

SUNJUN KIM[*], Daegu Gyeongbuk Institute of Science and Technology, Republic of Korea

WENZHE CUI, Stony Brook University, USA

XIAOJUN BI, Stony Brook University, USA

ANTTI OULASVIRTA, Aalto University, Finland

Despite the prevalence of shape-writing (gesture typing, swype input, or *swiping* for short) as a text entry method, there are currently no public datasets available. We report a large-scale dataset that can support efforts in both empirical study of swiping as well as the development of better intelligent text entry techniques. The dataset was collected via a web-based custom virtual keyboard, involving 1,338 users who submitted 11,318 unique English words. We report aggregate-level indices on typing performance, user-related factors, as well as trajectory-level data, such as the gesture path drawn on top of the keyboard or the time lapsed between consecutively swiped keys. We find some well-known effects reported in previous studies, for example that speed and error are affected by age and language skill. We also find surprising relationships such that, on large screens, swipe trajectories are longer but people swipe faster.

CCS Concepts: • **Human-centered computing** → *Interaction design process and methods*; • **Information systems** → **Database design and models**.

Additional Key Words and Phrases: Text Entry; Swiping; Shape-writing; Gesture Typing; Phrase set; Dataset

**ACM Reference Format:**

Luis A. Leiva, Sunjun Kim, Wenzhe Cui, Xiaojun Bi, and Antti Oulasvirta. 2021. How We Swipe: A Large-scale Shape-writing Dataset and Empirical Findings. In *23rd International Conference on Mobile Human-Computer Interaction (MobileHCI '21), September 27-October 1, 2021, Toulouse & Virtual, France.* ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3447526.3472059

## 1 INTRODUCTION

Shape-writing, also known as gesture typing, swype input, swipe to text, or just *swiping* (for short), is a prevalent mobile text entry method currently supported by all mobile vendors. Contrary to regular touch typing, where the user touches one key at a time and lifts up the finger to enter one character, swiping is a *word*-based text entry method: The finger lands on (or close to) the first key of the desired word and then, without lifting the finger from the keyboard, it traverses (the vicinity of) all the keys until reaching the last character of the word, generating a trajectory of touch points as a result. See Figure 2 for some examples.

While original studies suggested that swiping is faster than touch typing [25], recent empirical studies have challenged this assumption [34], calling for more research on the involved user factors and dynamics. However, until now, swiping performance has been researched in small-scale experiments [39], in large-scale studies as one of many text methods [34], or using proprietary, undisclosed datasets [37]. At the moment, there is no publicly available dataset where researchers

[*] Work done in part while affiliated with Aalto University.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2021 Copyright held by the owner/author(s).

Manuscript submitted to ACM

could access raw movements dynamics such as the gesture path drawn on top of the keyboard or the time lapsed between consecutive swiped keys. Collecting such data is challenging, because most mobile keyboards are vendor-locked and do not offer an API for collecting such data.

This paper contributes to efforts in understanding typing performance with mobile devices, a central topic in HCI; see e.g. [3, 4, 21, 36, 41]. We present a new dataset – the first of its kind – for conducting research on mobile swiping together with first observations of correlates of typing performance. To improve text entry techniques, it is important to understand their effects beyond controlled laboratory studies. While most studies in HCI have involved a relatively low number of participants [10], and often focused on prototype evaluation, we report here results from a large-scale dataset of over 1,300 volunteers. Large-scale analyses of mobile interaction are relatively rare and mostly undertaken by commercial organizations that hold the datasets proprietary. There are notable exceptions [9, 13, 21, 34] but they did not investigate swiping behavior. The analyses presented in this paper can contribute to improve our current understanding swiping performance and serve as training data for machine learning models.

To this end, we first present the data collection method and then describe our dataset. Next, we report on distributions of commonly used metrics of typing performance, including words per minute (WPM) and word error rate (WER). Then, to better understand mobile swiping behavior, we present observations on the effect of demographic factors and typing styles on performance. Our dataset and accompanying software are publicly available (see Appendix).

In the analysis of the dataset, we make several interesting findings, among which we highlight the following ones:

(1) Swiping with the thumb is faster than with any other finger.
(2) People swipe faster on large screens and invest less reading effort.
(3) On large screens, swipe trajectories are longer but swipe times are shorter.
(4) The more the user is familiarized with swiping, the higher the word error rate.
(5) Native English speakers swipe random words slower than non-natives.
(6) Text entry speed and word errors are affected by age, swipe hand, and language skill.

## 2 RELATED WORK

Invented by Zhai and Kristensson in 2003 [48], swiping has become a widely adopted text entry method on mobile devices.[1] It well suits touch-based interaction, and relaxes the requirement of precisely acquiring a small key on a soft keyboard. To date, this text entry method is supported by major commercial soft keyboards including Google's Gboard, Microsoft' SwiftKey, and iOS's built-in keyboard on iPhones.

The research community has also carried out a large amount of research on swiping techniques. For example, swiping has been extended to support mid-air text entry [31], eyes-free input [50], ring-based input [20], phone-tilting based input [46]. In addition to text entry, swiping paradigm has also been extended to support command input [1, 11, 26]. A user enters a command by gesturing a shortcut related to the command name. Such a method shows advantages in learnability compared with hotkey-based command input [11]. The swiping dataset reported in this paper will well serve these gesture typing based input methods, as it provides data for understanding fine-grained swiping behavior, and will serve as a basis for training gesture typing algorithms.

[1] The first publication on shape-writing technology is Kristensson's master's thesis from August 2002.

### 2.1 Phrase sets

The importance of representative phrase sets in studies of text entry has been acknowledge in extant literature. In the past, researchers used ad-hoc text sources for their experiments, such as sentences drawn from a Western novel [23], quotations from Unix's `fortune` program [22], news snippets [49], street addresses [19], or passages from Sherlock Holmes [40] and Alice in Wonderland [42]. Using ad-hoc, proprietary text sources is often considered a bad practice because text entry studies could not be accurately reproduced. To help the situation, researchers have proposed automated methods to create phrase sets [17, 28, 33]. MacKenzie et al. [30] released a phrase set consisting of 500 English idioms, and Vertanen et al. [43] released the EnronMobile phrase set, including empirical data regarding sentence memorability. Both Mackenzie's and EnronMobile phrase sets are today the most popular ones in text entry experiments. Kristensson et al. [24] compared both phrase sets and found not much difference between them, although the actual differences are *conceptually* rather large. For example, EnronMobile is better suited to evaluating mobile text entry methods, as it contains genuine mobile emails. As explained in Section 3.3, we will use this dataset as a baseline and will collect another set of more challenging words to ensure that our final dataset is diverse and representative of the English language.

### 2.2 Large-scale text entry studies

Text entry studies are mostly conducted in laboratory settings, mainly because of the difficulty in controlling the highly variable environment and reaching a larger audience. Despite the challenges, some larger-scale studies have been published, pursuing higher external validity. Previous large-scale studies tried different approaches for data collection, mainly simulation-based and on-line study methods.

Simulation-based methods enable a large-scale automated and repetitive analysis of a text entry system without performing an actual study. For example, Octopus [7] tried a simulation-based approach utilizing a collection of real-user text entry data. The resulting data comprise simulated touchpoints, which can operate the soft keyboard. A model-based approach is also possible. For example, Fowler et al. [16] implemented a human-like typing model, which simulates noisy touches, and Quinn and Zhai [37] modeled the gesture typing stroke generation process.

On-line data collection methods have been investigated to reach a larger audience. This kind of method involves real users and maximizes external validity. However, conducting such a study is costly, and a few challenges followed because of the method's uncontrolled nature. For example, 1) the apparatus varies between the participants, 2) the internal state of the system is unknown, 3) it exhibits a high drop-out rate [34], and 4) the data contains a significant amount of exceptions. The first and second issues could be partially solved by deploying a dedicated keyboard app to the public, like in the study by Reyal et al. [39]. They evaluated both the regular keyboard input and gesture typing. However, forced app installation increases the hurdle for study participation. A fully web-based study is an alternative. Two notable large-scale data collection studies [13, 34] have been performed recently, using physical keyboards [13] and built-in soft keyboards [34]. Because the keyboard's internal state (or text entry system) is unknown, the following workaround was suggested: By monitoring the input field changes, an estimation algorithm classifies the input events [2, 34]. The third and fourth issues (high drop-out rates and invalid trials) should be treated with a careful preprocessing of the data.

## 3 DATA COLLECTION METHOD

In the following we describe key design objectives of our work and the choices that support them. The first key objective is to collect a rich dataset on swiping, covering both movement-level and task performance-level data, since currently there is no such dataset in the research literature. The second objective is a large and representative sample of swiped words. Capturing natural variability in swiping is important for more realistic empirical analyses as well as for training machine learning techniques. A third key objective is to provide first larger-scale observations about swipe-based typing by cross-correlating it against user-related factors, such as demographics, language skill, as well as strategies like the choice of input finger.

We created the website https://swipetest.aalto.fi where users can test their shape-writing performance and compare to others. We contacted TypingMaster, a company that offers typewriting courses online and has a large active user base. The company advertised our test on their website for three months so that we could reach an international audience. HTTP requests to the site that originated from devices detected as mobile (screen width < 600 px) were redirected to our test. We also advertised the test on social media (Twitter and Facebook). There was no financial compensation for the users who took the online test, which triggered their intrinsic motivation to test their swiping efficiency.

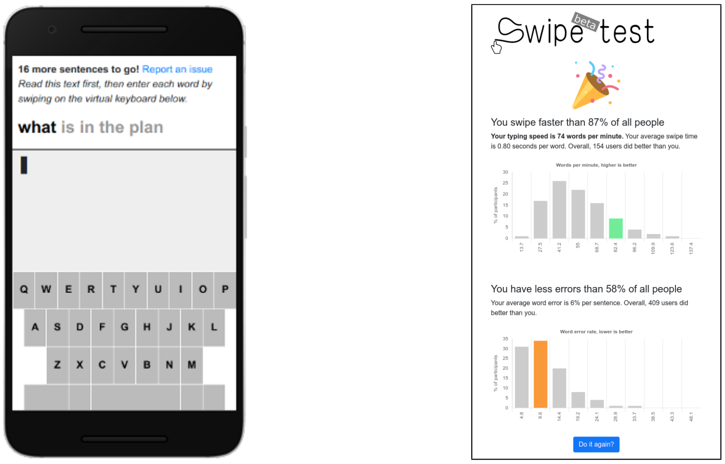
A web-based method, as opposed to a laboratory or app-based data collection, allows for a larger sample and broader coverage of different mobile devices, but comes with the caveats of self-selection and compromised logging accuracy. Still, the typing test setting imposes a more controlled environment than an in-the-wild study [34]. We ensured our test site supports all major mobile browsers, making it responsive to different screen sizes; see Section 3.1. Finally, we should remind the reader that our study is actually a data collection experiment rather than a real mobile typing test. For this reason, our virtual keyboard does not offer intelligent text entry assistance such as word autocompletion or realtime word suggestions, nor a working statistical decoder.

### 3.1 Soft keyboard development

We developed a JavaScript application that renders a virtual QWERTY keyboard on a `canvas` element. The keyboard layout looks the same for all users and it is programmatically adapted to the available screen size of the device, ensuring a consistent aspect ratio ($H/W = 1.4$ obtained by averaging 32 keyboard apps on iOS and Android markets). Note that for swiping the `space` key becomes unnecessary because all keyboards automatically insert a space after each swiped word. However, we decided to add it together with other 'dummy' keys (with no associated characters) in order to render a keyboard that would look familiar to the participants. Tapping or swiping over dummy keys has no effect on the produced text.

Figure 1(a) depicts the virtual keyboard as shown on an Google Nexus 6 phone. In our keyboard, while swiping, the finger trajectory is overlaid on top of the keys and gradually fades out, as currently implemented by all major mobile keyboard vendors. We note that the sampling resolution of the captured JavaScript touch events is around 16 ms (60 Hz), as measured by the logged touch event timestamps, indicating that modern mobile browsers can provide fine-grained movement data.

As hinted before, the virtual keyboard does not include statistical word decoding capabilities. This is to deconfound the effect of the decoder from user performance. Instead, the keyboard records any swipe trajectory and verifies that the touch coordinates are articulated around (either inside of or close to the vicinity of) the expected keys. For example, following the example in Figure 3, a swipe for the word "`what`" should begin around the "`w`" key, then move around the "`h`" key, then move around the "`a`" key, and finally end around the "`t`" key. To decide what is 'close enough', we empirically set a threshold radius of $r = 1.5 \max(h, w)$ px where $w$ and $h$ are the width and height of each key, respectively. Then, we compute the (ordered) sequence of traversed keys, removing duplicates, and verify that it matches the sequence of characters in the prompted word. This process is illustrated in Figure 3(b).



**16 more sentences to go!** Report an issue

*Read this text first, then enter each word by swiping on the virtual keyboard below.*

**what** is in the plan

Swipe test

You swipe faster than 87% of all people

**Your typing speed is 74 words per minute.** Your average swipe time is 0.80 seconds per word. Overall, 154 users did better than you.

You have less errors than 58% of all people

Your average word error is 6% per sentence. Overall, 409 users did better than you.

(a) (b)

Fig. 1. Left: Study prompt, with the deployed virtual keyboard at the bottom. Right: landing page shown to one of the participants at the end of the study.

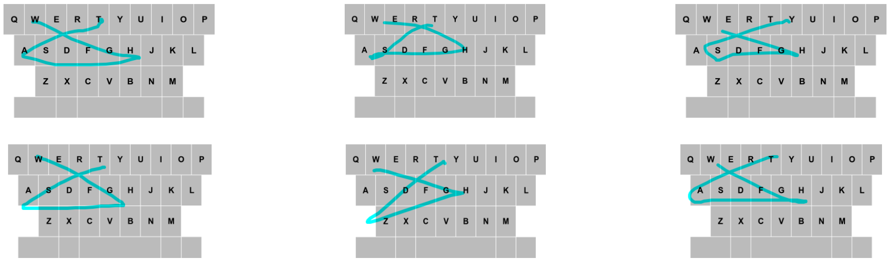
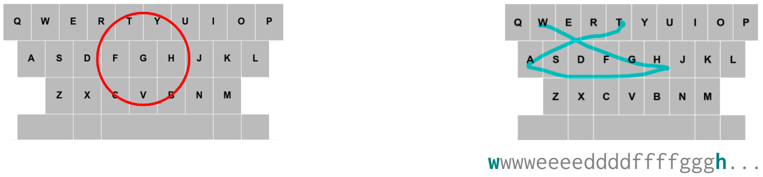


Fig. 2. Example of different swipes of the same word entered by different participants.



(a) Touch key tolerance

(b) Swipe trajectory for the word 'what'

Fig. 3. To verify whether a swipe trajectory followed the expected sequence of keys, we set a threshold radius of 50% larger than the current key size, shown in (a) around the letter "g", and computed the resulting sequence of traversed keys for each touch point (b).

### 3.2 Procedure

The study follows a procedure used in most text entry studies [39]; that is, a *transcription task* where an English phrase is presented and it must be entered using swiping. Our online test prompts each participant with 16 short sentences, the first of which is a warm-up sentence that is not logged.

Before starting the test, participants had to fill out a short questionnaire where they indicated some demographic information (e.g. age, gender, nationality, English level, etc.) together with skill-related information (e.g. swipe use, dominant hand, swipe finger, etc.). See Figure 9 for some examples of the collected data. Participants also had to acknowledge that they had read the instructions and gave their consent for data collection. Upon submitting these initial data, the first sentence was displayed.

Each participant had to swipe all the words for a given sentence (as fast and as accurately as possible) in order to advance to the next sentence. A counter in the top part of the screen indicates the number of remaining sentences to finish the test; see Figure 1(a). Each sentence was visible at all times. Breaks could be taken between the sentences. We used the following color hints to help the participants keep track of their progress. The word to be entered is rendered in bold typeface. Whenever a word if successfully entered, the word becomes blue. Pending words are shown in gray color. Finally, if the word is swiped wrongly, it becomes red and the participant has to swipe it again, until getting it right. In sum, there is no room for uncorrected errors in our study.

Upon entering all the sentences successfully, a summary page with performance statistics was shown to the participant. Figure 1(b) shows an example of such a landing page. Similar to Palin et al. [34], we chose well-understood performance metrics covering speed and errors and updated performance feedback only after participants completed the test. Participants could redo the test, if desired, to get a better estimate of their performance. Every time a user access the test there will be different sentences.

### 3.3 Phrase sets

We compiled two different phrase sets for our study. On the one hand, we created a set of 4-words sentences on the fly, drawn from 4 different word lists that we describe below. On the other hand, we used the memorable sentences from the EnronMobile dataset [43], a curated set of 200 short sentences written on Blackberry devices. This phrase set has been extensively used in mobile studies and it is widely considered as representative of the English language [34, 43]. We excluded long sentences (having more than 8 words) from this set to make our two phrase sets comparable, as sentences in mobile text entry evaluations are considerable short, around 9 words/sentence [43]. We concluded to a set of 169 sentences in total. We will refer to the first phrase set as `Random` and the second phrase set as `Enron`. Table 1 summarizes both sets.

To create the `Random` phrase set, we downloaded (1) the 10,000 most common English words according to Google's Trillion Word Corpus[2] and (2) the Forbes 2019 Global 2000 list,[3] which ranks the top public companies in the world. We created a word list from the Forbes companies by splitting each company name by the space delimiter. In sum, the `Random` phrase set includes:

(1) Highly frequent words: top 2k words from the common English list.
(2) Common words: next best 3k words from the common English list.
(3) Infrequent words: remaining 5k words from the common English list.
(4) Out-of-vocabulary (OOV) words: words from the Forbes Global 2000 and the common English list that are not in the English dictionary.[4]

By using two different phrase sets, we can verify how fluent and consistent users are while swiping. For example, words from the `Enron` sentences should be easier to swipe, since they are everyday words. One third of the sentences

[2] https://github.com/first20hours/google-10000-english
[3] https://www.forbes.com/global2000/
[4] We used the `/usr/share/dict/words` dictionary, available in all Unix systems.

| Set | Sentences | Running words | Words/Sentence | Chars/Word |
|---|---|---|---|---|
| Enron | 169 | 796 | 4.71 ± 1.3 | 3.61 ± 1.7 |
| Random | N/A | 12213 | 4.00 ± 0.0 | 6.58 ± 2.5 |

Table 1. Statistical summary of our phrase sets.

presented to every participant are drawn from the Enron dataset. The remaining of the presented sentences are 4-words sentences composed on the fly from the Random phrase set. All 4-words sentences include a highly frequent word, a common word, an infrequent word, and an OOV word. Each of these 4 words are randomly selected. There are 236 unique common words in both phrase sets.

All sentences shown to the participants are lowercased with no punctuation symbols. We also remove 1-character words (e.g. "x", "e") from our Random phrases, since at least 2 distinct characters are needed for swiping on a keyboard. Swiping over the same key was supported by our virtual keyboard, though, as a few Enron sentences include 1-character words (e.g. "i like it"). We did not remove offensive or swear words, in order to ensure ecological validity of our data, i.e. our dataset accounts for real-world words, including both formal and informal language. Table 2 shows some of the sentences prompted to the users.

| Phrase set | Sample sentence |
|---|---|
| Enron | i was planning to attend<br>you can talk to becky<br>this seems fine to me<br>not even close |
| Random | offerings viewpicture long eagle<br>release vcr dodge visa<br>layer prefers hyundai definition<br>attack cube value link |

Table 2. Sample sentences from our phrase sets.

### 3.4 Logging

We logged the following swipe-related data: event timestamp, x and y coordinate, touch radius (in x and y axes), and rotation angle. This information is captured via JavaScript using standard browser APIs. We also logged the keyboard size, the prompted word, and whether it was swiped correctly or not. On the other hand, we logged the following user metadata: swipe use, age, gender, nationality, browser language,[5] device pixel ratio, screen size (height and width), English level, dominant hand, swipe hand, swipe finger, and mobile vendor. Screen size was measured in CSS pixels by the JavaScript engine of the web browser,[6] which are normalized pixels and do not depend on the physical screen density of the device. This way we ensure that the recorded screen sizes are comparable across devices and vendors.

The data were logged in a backend server written in PHP, using a JSON-based database format to store the users' metadata and plain text format to store swipe log files. We post-processed the data both at the sentence and word level, to ease analysis efforts for the interested researchers. Figure 9 provides concrete examples of the log files we collected.

[5] Distilled from the list of languages in the HTTP Accept-Language header. We only considered the top (preferred) language from that list.

[6] https://www.quirksmode.org/blog/archives/2010/04/a_pixel_is_not.html

Our dataset comprises 8,831,733 touch points corresponding to 11,318 unique English words swiped by 1,338 users. There are 11,295 unique words correctly swiped and 3,767 words wrongly swiped. Note that the set of failed words is a subset of the successfully entered words, since the participants had to re-enter a failed word in order to get the sentence right and advance to the next one. We did not control for who accessed our test, since it was made globally available, nor how long breaks happened between sentences, so a data cleaning step becomes necessary. For example, some users accessed our test for the sake of curiosity and did not finish it. This is rather common in online studies [13, 34, 38]. Only 398 users (30%) completed all 16 sentences, which is a comparable dropout rate to Palin et al.'s study [34]. Therefore, since our main focus is on data collection, we consider for analysis the users who entered at least 5 sentences, which comprises one third of all the requested sentences. Then, we programmatically excluded users who did not use a real mobile phone according to the max. number of touch points supported by the user agent[7] (should be greater than one) and the viewport size[8] (should be less than 600 px width and less than 900 px height, respectively). A more detailed analysis of these exclusions suggested that these 'fake mobile' users (33 cases) accessed our test in mobile simulation mode, which is available in all major browsers.

## 4 DATA ANALYSIS

Here we describe the participants considered for analysis, define our set of text entry metrics to investigate swiping performance, and describe the statistical decoder we developed.

### 4.1 Participants

Our final user sample accounts for 909 participants (598 female, 304 male, 7 other) who used either Android (647) or iPhone (262) smartphones. We will release our full dataset with 1,338 users, so that researchers can use it at their own discretion. Figure 4 summarizes the user sample.

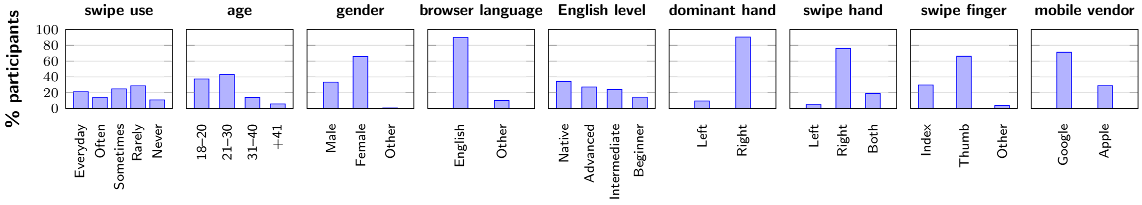


Fig. 4. Demographics characteristics of our analyzed user sample ($\boldsymbol{N} = \mathbf{909}$ participants).

Figure 5 shows the user population proportional to the contributions of each country. As can be shown, our collected user sample is rather diverse although a bit over-represented by the US (356 users); e.g. 106 users were from Philippines, 84 from Mexico, 81 from India, and 53 United Kingdom. To ease later analyses, considering that all words were in English and that most participants were from the US, we will split the nationality of our participants into US (356) and non-US (553), and the browser language will be split into English (815) and other (94).

Gender-wise, our sample is biased towards female users (598). Age-wise, our sample is biased towards young users: The average age of our participants is 25 years (Mdn=23). In the following we will use three age groups for analysis: youth (<20 years, 261 users), young (20–30 years, 440 users), and adult (+30 years, 179 users).

[7] https://www.w3.org/TR/pointerevents/
[8] https://mediag.com/blog/popular-screen-resolutions-designing-for-all/

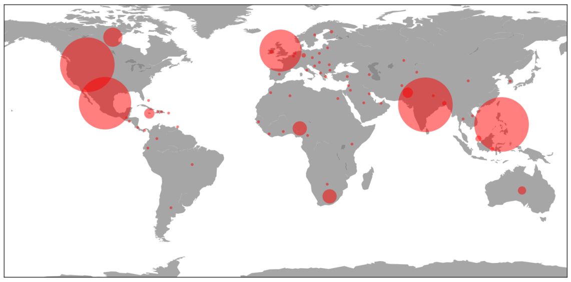


Fig. 5. Geographical distribution of our analyzed user sample. Each circle is proportional to the number of users from each country up to a saturation radius of 100 px to avoid visual clutter.

Regarding swipe familiarity, 193 users indicated they use it every day, 226 sometimes, 130 often, 261 rarely, and 99 never. Most users were right-handed (822 users) and were proficient in English: 312 declared to be native speakers, 248 had advanced knowledge, 218 had intermediate knowledge, and 131 were considered beginners. Most of the users swipe with their right hand (691 users) and 174 users indicated they use both hands. Also, most of the users swipe either with their thumb (601 users) or index finger (271). Finally, most participants used Android smartphones (647) and 372 users had a large phone screen (width >400 px).

### 4.2 Metrics

We report standard performance metrics in text entry (WPM, WER) and compute related metrics for swipe data, as discussed below. Further analysis was conducted using these measures, computed per sentence or word (where suitable) and aggregated per user. On the one hand, we consider the following sentence-level metrics:

**Words per minute (WPM)** is computed as number of characters in the sentence divided by five[9] divided by the time elapsed between the first touch point of the first entered word and the last touch point of the last entered word.

**Word error rate (WER)** is computed as the number of wrongly entered words according to a statistical decoder we describe in Section 4.3, divided by the number of words in the reference sentence.

**Interval time** is computed as the time elapsed between two consecutively swiped words (the difference between the time of the last touch point of the previous word and the time of the first touch point of the current word) and serves as a proxy for reading effort [18].

On the other hand, we consider the following word-level metrics, inspired by existing accuracy measures for evaluating computer pointing devices [29], whose purpose is to compute swiping performance via word difficulty estimators:

**Swipe length** is computed as the cumulative Euclidean distance between consecutive touch points in the swipe trajectory.

**Swipe time** is computed as the time elapsed between the first and last touch points of the swipe trajectory.

**Swipe error** is computed as the Dynamic Time Warping (alignment score) between the produced swipe sequence and the 'ideal' swipe sequence, or sokgraph [25].

We should note that, as discussed in Section 3.1, our virtual keyboard used a rather permissive procedure to not frustrate the users and allow them advance to the next word or sentence, while ensuring that the entered swipe

[9] http://www.yorku.ca/mack/RN-TextEntrySpeed.html

trajectory was not egregiously far off the expected swipe trajectory. Therefore, we will report WER post-hoc, according to a state-of-the-art statistical decoder, since in our study all logged sentences were eventually entered correctly.

### 4.3 Statistical decoding

We developed a word decoder following the principles outlined in SHARK2 [25]. The input of the decoder is a gesture trace $g$ on the keyboard, and the output is an $n$-best list of words from the vocabulary $W$. The word at the top of this list $\hat{w}$ is the most probable word. The vocabulary implemented in our decoder is the one described in Section 3.3.

The probability of decoding a word $w$ given a gesture trace $g$ on the keyboard is given by the Bayes theorem:

$$P(w|g) = \frac{P(g|w)P(w)}{P(g)} \tag{1}$$

where $P(g|w)$ is the decoding probability, given by a gesture model, $P(w)$ is the prior probability of the word, given by a language model, and $P(g)$ is a constant factor that does not influence on the decoding result. Indeed, the most probable word is given by

$$\hat{w} = \arg\max_W P(w|g) = \arg\max_W P(g|w)P(w) \tag{2}$$

because the $\arg\max$ operation is monotonous and so $P(g)$ can be ignored. This reasoning applies to any ranking function over the word vocabulary $W$.

As observed in Equation 2, the decoder comprises two components: a *gesture model* and a *language model*. For the gesture model we use elastic matching, parameterized by a Gaussian distribution [25]. The gesture likelihood is given by:

$$P(g|w) = \frac{P_s(g|w)P_l(g|w)}{\sum_{i \in W} P_s(g|i)P_l(g|i)} \tag{3}$$

where $P_s(g|i)$ and $P_l(g|i)$ are the probability from the shape and location channel. The shape channel evaluate gestures based on the shape information. The location channel examines the absolute location of the user's gesture trace on the keyboard. The details can be found in [25].

For the language model, we have considered both unigram and bigram models. For the unigram language model, we used 1/3 million most frequent English words and their frequencies [32]. Formally, the word probability in the unigram language model is given by their frequency:

$$P(w) = \frac{|w|}{\sum_{i \in W} |i|} \tag{4}$$

where $|\cdot|$ denotes the word frequency.

For the bigram language model, we used the Corpus of Contemporary American English (COCA) [12] comprising the data from 2012 to 2017, which contains over 5 million sentences. As a matter of fact, the Enron corpus was released in 2004 [43], so it is unlikely that our COCA partition may contain any Enron sentence. Formally, the word probability in a generic ($n$-gram) language model is given by their conditional frequency, i.e. a probability distribution over sequences of words:

$$P(w_1 \cdots w_m) = \prod_{i=1}^{m} P(w_i \mid w_1 \cdots w_{i-1}) \approx \prod_{i=1}^{m} \frac{|w_{i-(n-1)} \cdots w_i|}{|w_{i-(n-1)} \cdots w_{i-1}|} \tag{5}$$

where $m$ is the sentence length (word count). In the bigram language model, $n$ is set to 2, since it considers one context word.

*4.3.1 Effects of decoding.* Figure 6 reports decoding results considering the contribution of the decoder components (namely: no language model, unigram, and bigram language model) and using either top-1 or top-4 decoding accuracy.

While top-1 decoding accuracy is the most conservative decoding estimation, top-4 accuracy is based on the fact that modern soft keyboards show the decoded word together with three suggestions, and thus represents a more practical decoding estimation.

When no language model is used, the decoder relies on the gesture model only; see Equation 2. In this case, the WER reported below is attributed to the gesture decoding error alone; see 'Gesture' distributions. The unigram language model considers the probability of each word independently, i.e. with no context. Finally, the bigram language model considers the probability of each word conditioned on the previous word, for which we used the aforementioned COCA corpus. When any of these language models is used, WER is computed considering the contribution of both the gesture decoding and the language model. We remind that WER is computed as the number of wrongly decoded words divided by the number of words in the reference sentence.

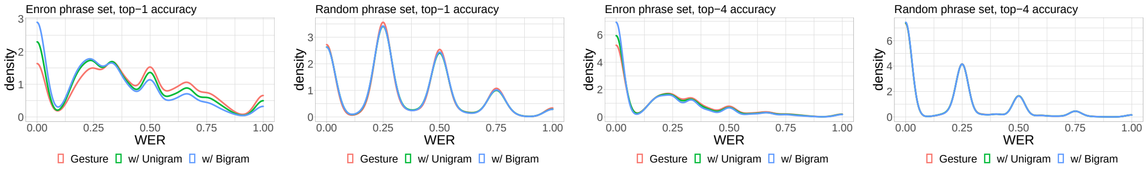


Fig. 6. Histograms of word decoding errors by phrase set and decoder components: Gesture model only (no language model), Gesture + Unigram language model, and Gesture + Bigram language model. Word error rates reported in [0,1].

Not surprisingly, it can be observed in Figure 6 that a language model improves decoding accuracy. Also, as expected, using top-4 accuracy improves the results considerably. We note, however, an observation about decoding. The word frequency in Random sentences is much smaller than words from Enron sentences, about one order of magnitude smaller, and so does their probability estimated by any language model. Therefore, Enron sentences do benefit from a bigram language model, as it is the one that achieves the best results. On the contrary, the bigram model is detrimental for the Random sentences, since those sentences were generated at random, and so word order does not matter. For this reason, it is the unigram language model the one that achieves the best results for Random sentences.

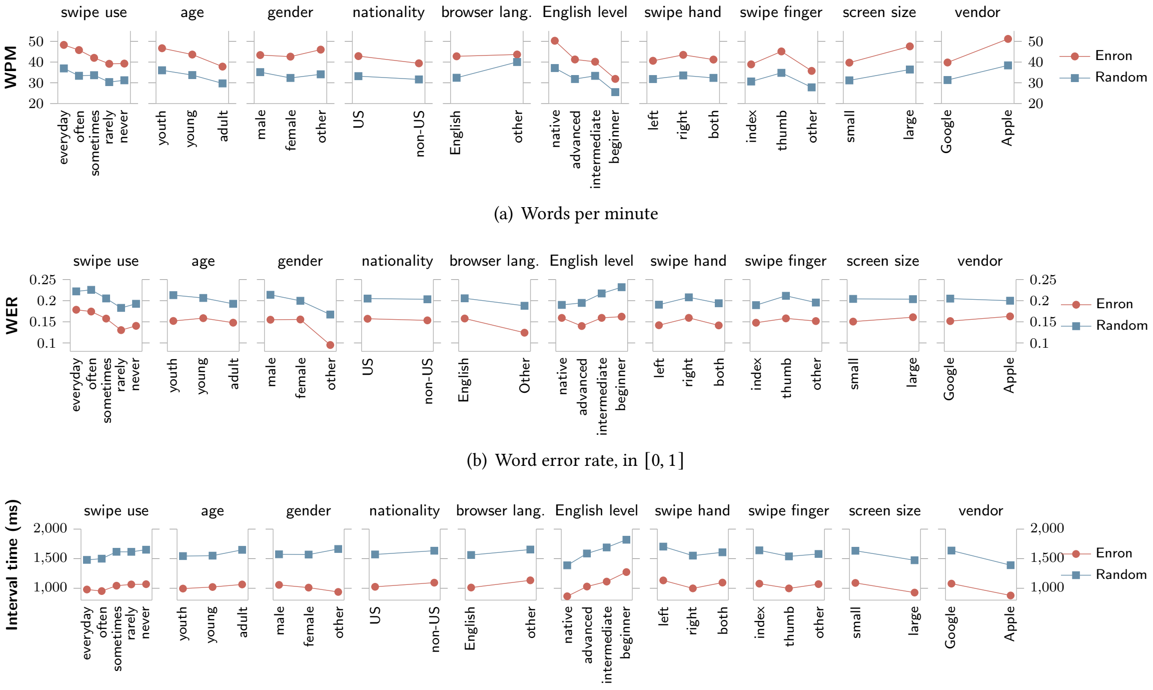
From now on we will use the combination of gesture model and unigram language model to compute WER, since it is the most common case: On the one hand, as previously mentioned, all major vendors display the top-1 recognized word in the soft keyboard's text field but also suggest up to three words. On the other hand, the unigram language model is the simplest model to create, since it is only necessary a list of words, without any context.

## 5 RESULTS

In the following we report overall performance estimates according to the metrics defined in Section 4.2. We then report on indicators of swiping performance and analyze the effect of demographics and other user-related factors. We aggregate the data by participant. Since most of our samples are not normally distributed, we use the Kruskal-Wallis test (non-parametric equivalent of the one-way ANOVA test) as Omnibus test. Then, if a significant difference is found and there is more than two conditions, we perform post-hoc pairwise comparisons using the Mann-Whitney U test with Holm-Bonferroni correction. Effect sizes are reported using $\phi$: A value of 0.1 is considered a small effect, 0.3 a medium effect, and 0.5 a large effect [8].

### 5.1 Sentence-level analysis

We begin by reporting sentence-level performance metrics. Later on we report word-level metrics.



(a) Words per minute

(b) Word error rate, in $[0, 1]$

(c) Interval time

Fig. 7. Mean distributions of sentence-level performance metrics.

*5.1.1 Words per minute.* Figure 7(a) reports text entry speed, in words per minute, for all considered factors. The average WPM is M=39.82 (SD=17.08) for the Enron phrase set, which is higher than the WPM observed in a similar web-based large-scale study of mobile typing behavior (36.17 WPM) that used the same phrase set [34]. The average WPM is M=31.11 (SD=12.49) for the Random phrase set. Overall, the fastest participant reached and average WPM of 72.98.

We did not find an effect of gender or browser language. All other comparisons were statistically significant; see Table 3. The effect size of the phrase set is moderate ($\phi = 0.271$), suggesting that Random sentences are slower to swipe, which is understandable since random words are longer (Table 1). The more familiarized is the user with swiping, the faster they enter text; c.f. 50 WPM (everyday use) vs. 40 WPM (never) for the Enron phrase set. Then, the older the user, the lower the WPM. People from US are significantly faster than others, and, surprisingly, those whose browser language is not English do swipe faster. This difference is larger for the Random phrase set, suggesting that native English speakers are biased towards entering common words; i.e. they require more reading time for random words because they do not see those often, whereas for non-natives, both common and unusual words look more similar. We discuss this further in Section 5.3. On the other hand, people who swipe with their dominant hand and the thumb are faster. We also observed that people swipe faster on larger screens (width >400 px) and on Apple devices.

*5.1.2 Word error rate.* Figure 7(b) reports text entry errors, expressed as word error rates, for all considered factors. The average WER is M=15% (SD=17.7%) for the Enron phrase set and M=15% (SD=17.5%) for the Random phrase set.

The median WER is 0% in both phrase sets, which reveals that our statistical decoder can recognize all entered words (per sentence) in most cases. We note that here we consider the top-4 words using Gesture + Unigram models; see Figure 6. We found a significant effect of swipe use, age, English level, and swipe finger. Among these, swipe use showed the largest differences. Surprisingly, the more familiarized is the user with swiping, the larger the WER. All other comparisons were non-significant; see Table 3. The effect size of the phrase set is non-significant, suggesting that both `Enron` and `Random` phrase sets are similar in terms of WER. However, we should remember that every random word that was not correctly swiped had to be re-entered until getting it right, which might have influenced the results of this analysis. Also notice the small range of the Y axis, which suggests that the our statistical decoder performs similarly across factors.

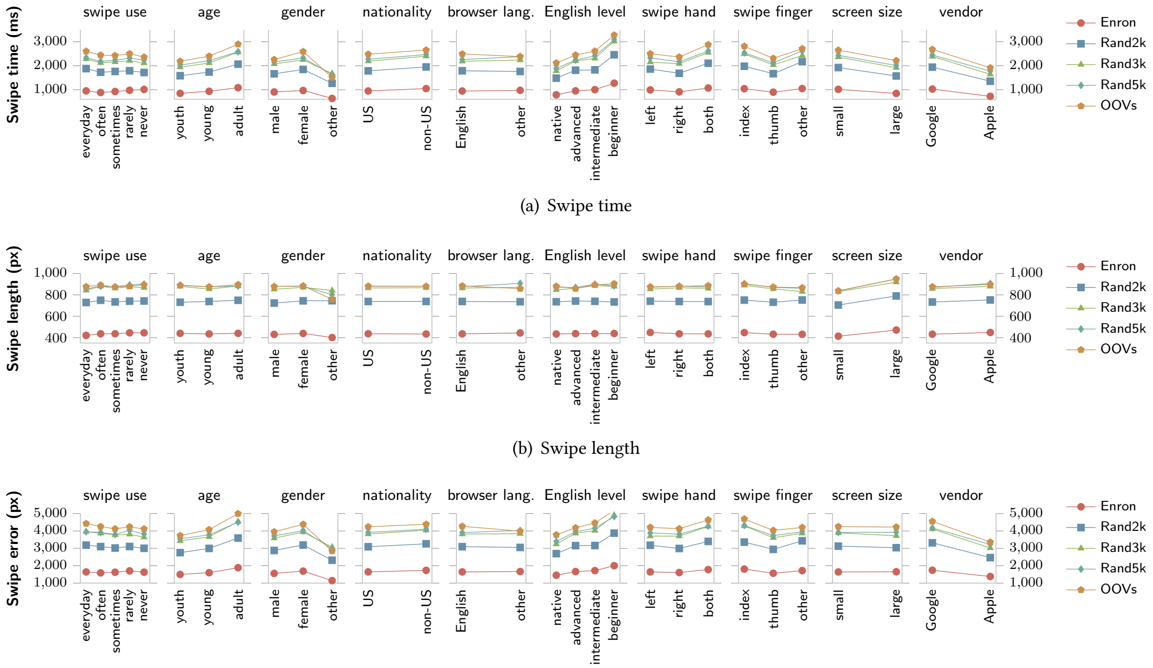
*5.1.3 Interval time.* Figure 7(c) reports the time elapsed between consecutive swipes for all considered factors. The average interval time is M=846.5 ms (SD=374.9 ms) for the `Enron` phrase set and M=1210.5 ms (SD=452.2 ms) for the `Random` phrase set. We did not find an effect of age, gender, browser language, or swipe finger. All other comparisons were significant; see Table 3. The effect size of the phrase set is large ($\phi = 0.445$), suggesting that sentences with random words require more reading effort. As expected, the more familiarized is the user with swiping, the higher interval times. This suggests that experienced users require less reading effort than swipe beginners. Non-US people require more thinking time, especially those with less English knowledge. Those who swipe with their non-dominant hand of the index finger require more interval time, presumably because of the influence of grip posture and motor control factors: People who swipe with their thumb are also holding the mobile phone with the same hand, which requires less effort to reach the screen. Interestingly, the larger the screen the smaller the interval times. We attribute this difference to the fact that on smaller screens it is more difficult to read texts, which eventually increases the time to between words. Also interestingly, Apple users required less interval time, which might be explained by other factors such as age or English knowledge. Indeed, if we split the data by these factors we can see that Apple users are younger (Mdn=20 years) than Google users (Mdn=25 years) and have more English natives (78% vs 56% of the users, respectively).

| **Factor** | **Levels ($g$)** | **WPM** | | | **WER** | | | **Interval time** | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $\chi^2$ | $\phi$ | | $\chi^2$ | $\phi$ | | $\chi^2$ | $\phi$ | |
| Swipe use | 5 | 28.60 | 0.126 | *** | 21.08 | 0.108 | *** | 17.83 | 0.099 | ** |
| Age | 3 | 26.90 | 0.122 | *** | 7.64 | 0.065 | * | 5.85 | 0.057 | |
| Gender | 3 | 2.54 | 0.037 | | 0.73 | 0.020 | | 2.35 | 0.036 | |
| Nationality | 2 | 130.38 | 0.268 | *** | 2.85 | 0.040 | | 84.63 | 0.216 | *** |
| Browser language | 2 | 2.40 | 0.036 | | 0.68 | 0.019 | | 5.20 | 0.054 | * |
| English level | 4 | 212.02 | 0.342 | *** | 7.96 | 0.066 | * | 168.03 | 0.305 | *** |
| Swipe hand | 3 | 17.87 | 0.099 | *** | 1.91 | 0.032 | | 12.36 | 0.083 | ** |
| Swipe finger | 3 | 25.67 | 0.119 | *** | 6.04 | 0.058 | * | 6.73 | 0.061 | * |
| Screen size | 2 | 48.94 | 0.164 | *** | 0.71 | 0.020 | | 45.15 | 0.158 | *** |
| Vendor | 2 | 149.68 | 0.287 | *** | 1.44 | 0.028 | | 78.38 | 0.208 | *** |
| Phrase set | 2 | 133.51 | 0.271 | *** | 0.16 | 0.009 | | 359.14 | 0.445 | *** |

Table 3. Statistical tests $\chi^2(g-1, N=1816)$ of sentence-level metrics and effect sizes $\phi$. Statistical significance is denoted as follows: $p < .001$ (***), $p < .01$ (**), $p < .05$ (*).

## 5.2 Word-level analysis

We now report word-level performance metrics, for which we plot the `Random` phrase set according to each of the four word categories considered: highly frequent words (Rand2k), common words (Rand3k), infrequent words (Rand5k), and out-of-vocabulary words (OOVs).



(a) Swipe time

(b) Swipe length

(c) Swipe error

Fig. 8. Mean distributions of word-level performance metrics.

*5.2.1 Swipe time.* Figure 8(a) reports the average swipe time for all considered factors. Overall, it is M=689.2 ms (SD=520.4 ms) for the Enron phrase set, M=1414 ms (SD=959.7 ms) for Rand2k words, and M=1933.2 ms (SD=1332.7 ms) for OOV words. We did not find an effect of browser language. All other comparisons were significant; see Table 4. However, in most cases the effect sizes are small, suggesting a small practical importance of these differences. The effect size of English level is moderate ($\phi = 0.245$), with native and advanced English speakers being considerably faster than beginners. There is a large effect size of word type ($\phi = 0.527$), however no differences were found between Rand5k and both Rand3k and OOVs. All other differences between word types were significant. This suggests that words belonging to the Enron phrase set are faster to swipe overall, and that Rand2k words are faster to swipe than any other word type in the Random phrase set.

*5.2.2 Swipe length.* Figure 8(b) reports the average swipe length for all considered factors. Overall, it is M=371.1 px (SD=256.6 px) for the Enron phrase set, M=687.3 px (SD=383.8 px) for Rand2k words, and M=799 px (SD=438.8 ms) for OOV words. We did not find an effect of age, gender, browser language, English level, or swipe hand. All other comparisons were significant; see Table 4. However, in most cases the effect sizes are small. Interestingly, there is a moderate effect size of screen size ($\phi = 0.195$), suggesting that swipe trajectories are significantly longer on larger screens, however this finding contradicts the previous evidence that users swipe faster on larger screens. We suspect that there are other demographic factors influencing this result. There is a large effect size of word type ($\phi = 0.681$), however no differences were found between Rand5k and both Rand3k and OOVs. All other differences between word types were significant. This suggests that words belonging to the Enron phrase set produce shorter swipe trajectories. and that Rand2k words are shorter than any other word type in the Random phrase set.

*5.2.3 Swipe error.* Finally, Figure 8(c) reports the average swipe error per word for all considered factors. Overall, it is M=1285.8 px (SD=922.4 px) for the Enron phrase set, M=2596.1 px (SD=1663.5 px) for Rand2k words, and M=3562.5 px (SD=2261.6 ms) for OOV words. We did not find an effect of browser language or screen size. All other comparisons were statistically significant; see Table 4. However, in most cases the effect sizes are small. There is a moderate effect size of mobile vendor ($\phi = 0.184$), but again we suspect it is because other demographic factors. There is a large effect size of word type ($\phi = 0.581$), however no differences were found between Rand5k and both Rand3k and OOVs. All other differences between word types were significant. This suggests that words belonging to the Enron phrase set are swiped more precisely, and that Rand2k words are swiped more accurately than any other word type in the Random phrase set.

| **Factor** | **Levels ($g$)** | **Swipe time** | | **Swipe length** | | **Swipe error** | |
|---|---|---|---|---|---|---|---|
| | | $\chi^2$ | $\phi$ | $\chi^2$ | $\phi$ | $\chi^2$ | $\phi$ |
| Swipe use | 5 | 30.42 | 0.082 *** | 17.01 | 0.061 ** | 15.93 | 0.059 ** |
| Age | 3 | 47.75 | 0.103 *** | 0.91 | 0.014 | 48.38 | 0.103 *** |
| Gender | 3 | 15.71 | 0.059 *** | 2.05 | 0.021 | 18.15 | 0.063 *** |
| Nationality | 2 | 168.67 | 0.193 *** | 6.15 | 0.037 * | 45.33 | 0.100 *** |
| Browser language | 2 | 0.10 | 0.005 | 0.02 | 0.002 | 0.57 | 0.011 |
| English level | 4 | 271.93 | 0.245 *** | 1.21 | 0.016 | 143.12 | 0.178 *** |
| Swipe hand | 3 | 21.44 | 0.006 *** | 0.92 | 0.014 | 16.06 | 0.060 *** |
| Swipe finger | 3 | 51.38 | 0.106 *** | 9.60 | 0.046 ** | 28.33 | 0.079 *** |
| Screen size | 2 | 67.11 | 0.122 *** | 173.38 | 0.195 *** | 0.18 | 0.006 |
| Vendor | 2 | 198.36 | 0.209 *** | 21.94 | 0.070 *** | 153.87 | 0.184 *** |
| Word type | 5 | 1257.72 | 0.527 *** | 2101.99 | 0.681 *** | 1530.50 | 0.581 *** |

Table 4. Statistical tests $\boldsymbol{X^2 (g-1, N = 4537)}$ of word-level metrics and effect sizes $\phi$. Statistical significance is denoted as follows: $\boldsymbol{p < .001}$ (***), $\boldsymbol{p < .01}$ (**), $\boldsymbol{p < .05}$ (*).

## 5.3 Summary of findings

In the following we discuss the most salient findings of our study:

*5.3.1 Swiping with the thumb is faster than with any other finger.* We were surprised to see that swiping with the thumb is faster than with the index finger, as the prototypical demonstration of swiping is performed with the index finger. Moreover, a previous study reported that reaching the corners of a mobile display is challenging with when using the thumb [6]. Comparing to one-finger tap typing, Azenkot and Zhai reported index finger typing is faster than thumb by 2.5 WPM [3], while Palin et al. [34] reported typing with the thumb is faster than with the index finger by a factor of 3 WPM in a larger scale study. Reyal et al. [39] also noted that swiping with the thumb is particularly effective in more mobile situations. It is possible, of course, that the thumb performance correlates with any of the other factors, such as age. Our dataset permits deeper analyses of the observation and in fact we plan to do so in future work.

*5.3.2 People swipe faster on large screens and invest less reading effort.* It is surprising that larger screens are associated with higher WPMs, while target selection performance generally degrades with smaller target size [35]. This may, like above, be confounded by other background demographics. However, one could also conjecture that it is associated with relatively smaller effects of motor noise. Likewise, we attribute the positive relationship of larger screen size with smaller between-word intervals to be because it is easier to read text on larger screens. A larger screen would also mean bigger keys and reduced finger occlusion, allowing users to prepare their swipe faster since they have a better view of the keyboard.

*5.3.3 On large screens, swipe trajectories are longer but swipe times are shorter.* We found that, on a large screen, swipe length is not only longer but also briefer. This could be related to the effects of motor noise, as above, but also because newer phones have better hardware which, in turn, may correlate with a more responsive touchscreen. This might correlate as well with pointing and steering laws: With larger screens, the target (i.e., key) is larger, thus easier to reach. Users can also speed up in the middle of the swipe trajectory, then slow down when the finger is close to the target [5]. In addition, on a smaller display the finger is more likely to cut across unwanted characters. Users may be mitigating that by regulating speed–accuracy trade-off such that they slower down writing on a smaller display.

*5.3.4 The more the user is familiarized with swiping, the higher the word error rate.* Somewhat paradoxically, user's swiping experience was associated with slightly higher WER, although the effect size is small ($\phi = 0.108$). Again, there could be latent effects of age or display size. However, it could also reflect *adaptation*: users experienced with a particular gesture decoder may have been penalized by the one we used; as switching to a similar text entry system (but still a different system) often causes breakdowns in learning curves, which might explain their higher swipe errors. We can further relate this finding to previous work in stroke gesture articulation that found expert users to be sloppier than novices [45].

*5.3.5 Native English speakers swipe random words slower than non-natives.* Interestingly, non-natives were faster while swiping random words than English natives. We tentatively attribute this to motor learning: People used to type in English may have internalized motor patterns that reflect the statistical distribution of that language. Our non-native participants come from different language backgrounds, some of which may by chance have a better match with the language. Put differently, for non-natives, most English words have about the same articulation difficulty. Another hypothesis is that non-native users pay more attention to characters – not words. For a user like this, changing from real words to random words does not make a big difference.

*5.3.6 Text entry speed and word errors are affected by age, swipe hand, and language skill.* Previous work has found a non-linear negative influence of age [34] in mobile touch typing, which we now can corroborate it also occurs in swiping. Recent work by Findlater et al. [15] has found important age and motor ability correlates impacting input performance. In our data, more concretely, right-handed young English speakers usually swipe faster than any other user group. Further work is needed to understand these effects more deeply, since, as previously highlighted, text entry performance is a multidimensional problem and should be investigated as such.

## 6 DISCUSSION

We collected swipe data from thousands of volunteers using a web-based transcription task and a custom-made virtual keyboard. Even if online studies of self-selected volunteers do not permit a rigorous control,[10] large samples increase statistical power and yield better estimates and shapes of distributions. Previous work on gathering mobile typing data outside the traditional lab experiment has relied on crowdsourcing [24] or custom mobile apps [9, 21]. The only large-scale web-based mobile typing study that we are aware of was conducted by Palin et al. [34] but they did not collect trajectory-level data on swiping, since participants used the built-in soft keyboard of their smartphones. On the

[10] For example, we cannot guarantee that users provided their actual age or swiping finger. However, given the conditions of our study, we are confident our participants operated in good faith.

contrary, our deployed keyboard layout was programmatically adapted to the available screen of each user's device, ensuring a consistent aspect ratio and within the range of available keyboards on the market. We have analyzed the aspect ratios in our data and they are normally distributed ($p = .704$, Kolmogorov-Smirnov test) with M=1.41 and SD=0.17. Therefore, we can safely assume that most keyboards used by other users will not be noticeably off from our tested keyboard.

To the best of our knowledge, ours is the first public dataset of its kind, and on an unprecedented scale. Nevertheless, this comes with limitations. Generalizability of the user sample is an issue: our participants are likely exhibiting a self-selection bias due to the nature of the recruitment website, which is a typing test website. Many of our participants were young females from the US interested in typing. We acknowledge that this is not representative of the general population and might bias the data towards representing a western, young, more technology-affine group of people. Yet, based on previous large-scale text entry studies [9, 13, 21, 34] we argue that our findings would not be qualitatively different with a demographically more diverse sample. For example, as discussed in Section 5.3, Palin et al. [34] reported that typing with the thumb is faster than the index by 3 WPM, whereas we observed a difference of 4 WPM in our data. Furthermore, we were able to identify dominant user factors influencing swiping behavior, not previously reported elsewhere. Hopefully others will deploy our virtual keyboard in a different study setting and collect new data, contributing thus to expanding our current knowledge about mobile typing performance.

Given than were interested in collecting a rich dataset that is large and representative of the English language, the most straightforward way to encourage users to swipe as many words as possible is to create random phrases on the fly. Now that our dataset contains many real-word swiped words, text entry researchers can create their own phrases by recombining the collected data. This is a standard procedure in other machine learning areas such as handwriting transcription [27, 47].

We used a transcription task to assess typing performance which requires the participant to dedicate part of their attention to the transcribed sentence in addition to the entered text and the keyboard. Part of this attention is allocated to memorizing the prompted sentence to some extent, hence memorizing a grammatically-valid sentence is easier than memorizing random words. Alternative methods to assess typing performance include composition tasks [44] or even object-based methods, such as instructing users to annotate an image [14]. We see promising follow-up work in both changing the nature of the task and the parameters of individual tasks. For example, using our web-based test, we could investigate the effect of different composition tasks and individual task parameters, such as the effect of difficulty of a sentence set on transcription task performance. Such investigations are difficult to perform using traditional text entry experimental methods and we hope our dataset will be inspirational for other text entry researchers.

Finally, we should mention the absence of autocompletion and word suggestions in our deployed soft keyboard. This design choice was to remove any possible bias from a particular decoder implementation and to gauge the most natural word gestures that are not tuned to a particular recognizer's ability. We believe this should be seen as a benefit of our dataset, as by collecting algorithm-free data we can ensure they are not affected by any recognition or correction technique. Future work should investigate the influence of writing assistance methods, such as autocomplete and word suggestion, to consider the presence of all modern aspects of swiping technology. Furthermore, many more interesting insights could be derived from this rich dataset, including for example the potentially complex swiping patterns that have performance implications. For example, What is the gesture variability among users? Which kinds of gestures are more difficult? Where on the keyboard do people tend to make mistakes? We leave these questions as an opportunity for future work.

## 7 CONCLUSION

We have collected a large-scale dataset of mobile swiping behavior and have reported initial observations that can support efforts in both empirical study of swiping as well as the development of better intelligent text entry techniques. Our study allowed us to carry out detailed statistical analyses of swipe-based metrics and correlates of typing performance, including e.g. user demographics, English level proficiency, and mobile vendor. Our collected dataset is the first of its kind, in the sense that it comprises *raw* swipe trajectories, including touch points, timestamps, and the finger's area of contact with the touchscreen. The presented analysis confirms prior findings on text entry performance but also gives us new insights into the complex swiping behavior of people and the large variations between them.

Although the nature and scale of our work have surpassed its precedents in the text entry literature [9, 13, 21, 24, 34, 37, 39], many questions beyond the scope of this paper require further research. These include further, longitudinal empirical studies to disentangle confounds between the various factors we have considered for analysis. Further empirical investigation may refine out current knowledge about swiping behavior. To this end, we are releasing our dataset and associated software to assist further efforts in modeling, machine learning, and improvements of text entry methods.

## A THE HOW-WE-SWIPE DATASET

Our dataset is available at https://osf.io/sj67f/. It contains over 8M touch points corresponding to over 11k unique English words swiped by over 1,300 volunteers. The dataset includes all data reported in this paper, including demographics, swipe logs, sentence stimuli, and failed words. Figure 9 provides an example of the kind of data one might find in our dataset. The source code of our logging application is available at https://github.com/luileito/swipetest.

**Swipe log file example**

```
sentence timestamp keyb_width keyb_height event x_pos y_pos x_radius y_radius angle word is_err
force_instrumental_reliance_replacement 1576480557856 360 205 touchstart 140 100 1.33... 1.33... 0 force 0
force_instrumental_reliance_replacement 1576480557864 360 205 touchmove 140 100 1.49... 1.49... 0 force 0
[ more lines ]
```

**Processed user metadata**

```
age dominant_hand english_level familiarity gender language max_touch_points nationality \
screen_height screen_width swipe_finger swipe_hand timestamp uid vendor
19 right beginner sometimes female en 5 ae 640 360 thumb right 1583309231 0q1... google inc.
19 left native rarely female en 5 us 869 412 index right 1583136907 124... google inc.
[ more lines ]
```

**Processed sentence-level stats**

```
good_wpm_swipe sentence fail_wpm_classic fail_wpm_swipe fail_interval_time fail_length fail_time good_length  \
fail_dtw good_interval_time good_dtw good_time good_wpm_classic wer username dataset
18.1... they_have_capacity_now 15.9... 9.9... nan 1409.5... 3012.5 904.5... 10498.2... 1703.5 3660.1... 1648.0 20.0... 0.25 0q1... enron
39.2... what_a_jerk nan nan 453.5... nan nan 2146.8... 1528.0 35.3... 0.0 0q1... enron
[ more lines ]
```

**Processed word-level stats**

```
dataset dtw is_failed length sentence time username word
enron 4853.1... 0 1078.9... they_have_capacity_now 1709 0q1... they
enron 2467.1... 0 730.1... they_have_capacity_now 1587 0q1... have
[ more lines ]
```

Fig. 9. Dataset file samples. A backslash (\) denotes a line continuation. An ellipsis (. . .) denotes an intentional omission of some data, such as decimal places, for brevity's sake.

## ACKNOWLEDGMENTS

We thank Andreas Komninos and Byungjoo Lee for reviewing earlier drafts of this paper and TypingMaster for helping us reach a large audience of mobile users. We also thank our anonymous referees for their constructive feedback. This research was supported by the Academy of Finland (grant numbers 291556, 310947) and the DGIST Start-up Fund Program of the Ministry of Science and ICT (2021010011).

## REFERENCES

[1] J. Alvina, C. F. Griggio, X. Bi, and W. E. Mackay. 2017. CommandBoard: Creating a general-purpose command gesture input space for soft keyboard. In *Proc. UIST.* 17–28.

[2] A. S. Arif and A. Mazalek. 2016. WebTEM: A Web Application to Record Text Entry Metrics. In *Proc. ISS.* 415–420.

[3] S. Azenkot and S. Zhai. 2012. Touch Behavior with Different Postures on Soft Smartphone Keyboards. In *Proc. MobileHCI.* 251–260.

[4] N. Banovic, V. Rao, A. Saravanan, A. K. Dey, and J. Mankoff. 2017. Quantifying Aversion to Costly Typing Errors in Expert Mobile Text Entry. In *Proc. CHI.* 4229–4241.

[5] M. Beaudouin-Lafon, S. Huot, H. Olafsdottir, and P. Dragicevic. 2014. GlideCursor: Pointing with an Inertial Cursor. In *Proc. AVI.* 49–56.

[6] J. Bergstrom-Lehtovirta and A. Oulasvirta. 2014. Modeling the functional area of the thumb on mobile touchscreen surfaces. In *Proc. CHI.* 1991–2000.

[7] X. Bi, S. Azenkot, K. Partridge, and S. Zhai. 2013. Octopus: Evaluating Touchscreen Keyboard Correction and Recognition Algorithms Via "Remulation". In *Proc. CHI.* 543–552.

[8] M. Borenstein. 2009. Effect sizes for continuous data. In *The handbook of research synthesis and meta-analysis* (2nd ed.), H. Cooper, L. V. Hedges, and J. C. Valentine (Eds.). Sage Foundation, 221–235.

[9] D. Buschek, B. Bisinger, and F. Alt. 2018. ResearchIME: A Mobile Keyboard Application for Studying Free Typing Behaviour in the Wild. In *Proc. CHI.* 1–14.

[10] K. Caine. 2016. Local Standards for Sample Size at CHI. In *Proc. CHI.* 981–992.

[11] W. Cui, J. Zheng, B. Lewis, D. Vogel, and X. Bi. 2019. HotStrokes: Word-gesture shortcuts on a trackpad. In *Proc. CHI.* 1–13.

[12] M. Davies. 2018. The corpus of contemporary American English: 1990-present. (2018).

[13] V. Dhakal, A. M. Feit, P. O. Kristensson, and A. Oulasvirta. 2018. Observations on Typing from 136 Million Keystrokes. In *Proc. CHI.* 1–12.

[14] M. D. Dunlop, E. Nicol, A. Komninos, P. Dona, and N. Durga. 2015. Measuring inviscid text entry using image description tasks. In *Proc. CHI Workshop on Inviscid Text Entry and Beyond.*

[15] L. Findlater and L. Zhang. 2020. Input Accessibility: A Large Dataset and Summary Analysis of Age, Motor Ability and Input Performance. In *Proc. SIGACCESS.* 1–6.

[16] A. Fowler, K. Partridge, C. Chelba, X. Bi, T. Ouyang, and S. Zhai. 2015. Effects of Language Modeling and Its Personalization on Touchscreen Typing Performance. In *Proc. CHI.* 649–658.

[17] M. Franco-Salvador and L. A. Leiva. 2018. Multilingual Phrase Sampling for Text Entry Evaluations. *Int. J. Hum.-Comput. Stud.* 113, 1 (2018).

[18] D. R. Gentner, J. T. Grudin, S. Larochelle, D. A. Norman, and D. E. Rumelhart. 1983. *Cognitive Aspects of Skilled Typewriting.* Springer, Chapter A Glossary of Terms Including a Classification of Typing Errors, 39–43.

[19] I. E. González, J. O. Wobbrock, D. H. Chau, A. Faulring, and B. A. Myers. 2007. Eyes on the Road, Hands on the Wheel: Thumb-based Interaction Techniques for Input on Steering Wheels. In *Proc. GI.* 95–102.

[20] A. Gupta, C. Ji, H.-S. Yeo, A. Quigley, and D. Vogel. 2019. RotoSwype: Word-gesture typing using a ring. In *Proc. CHI.* 1–12.

[21] N. Henze, E. Rukzio, and S. Boll. 2012. Observational and Experimental Investigation of Typing Behaviour Using Virtual Keyboards for Mobile Devices. In *Proc. CHI.* 2659–2668.

[22] P. Isokoski and R. Raisamo. 2000. Device Independent Text Input: A Rationale and an Example. In *Proc. AVI.* 76–83.

[23] C.-M. Karat, C. Halverson, D. Horn, and J. Karat. 1999. Patterns of Entry and Correction in Large Vocabulary Continuous Speech Recognition Systems. In *Proc. CHI.* 568–575.

[24] P. O. Kristensson and K. Vertanen. 2012. Performance Comparisons of Phrase Sets and Presentation Styles for Text Entry Evaluations. In *Proc. IUI.* 29–32.

[25] P. O. Kristensson and S. Zhai. 2004. SHARK2: A Large Vocabulary Shorthand Writing System for Pen-based Computers. In *Proc. UIST.* 43–52.

[26] P. O. Kristensson and S. Zhai. 2007. Command strokes with and without preview: using pen gestures on keyboard for command selection. In *Proc. CHI.* 1137–1146.

[27] L. A. Leiva, V. Romero, A. H. Toselli, and E. Vidal. 2011. Evaluating an Interactive-Predictive Paradigm on Handwriting Transcription: A Case Study and Lessons Learned. In *Proc. COMPSAC.* 610–617.

[28] L. A. Leiva and G. Sanchis-Trilles. 2014. Representatively Memorable: Sampling the Right Phrase Set to Get the Text Entry Experiment Right. In *Proc. CHI.* 1709–1712.

[29] I. S. MacKenzie, T. Kauppinen, and M. Silfverberg. 2001. Accuracy Measures for Evaluating Computer Pointing Devices. In *Proc. CHI.* 9–16.

[30] I. S. MacKenzie and R. W. Soukoreff. 2003. Phrase sets for evaluating text entry techniques. In *Proc. CHI EA.* 754–755.

[31] A. Markussen, M. R. Jakobsen, and K. Hornbæk. 2014. Vulture: A Mid-air Word-gesture Keyboard. In *Proc. CHI.* 1073–1082.

[32] P. Norvig. 2019. *Natural Language Corpus Data.* O'Reilly Media, Inc., 219–242.

[33] T. Paek and B.-J. P. Hsu. 2011. Sampling Representative Phrase Sets for Text Entry Experiments: A Procedure and Public Resource. In *Proc. CHI.* 2477–2480.

[34] K. Palin, A. M. Feit, S. Kim, P. O. Kristensson, and A. Oulasvirta. 2019. How Do People Type on Mobile Devices? Observations from a Study with 37,000 Volunteers. In *Proc. MobileHCI.* 1–12.

[35] P. Parhi, A. K. Karlson, and B. B. Bederson. 2006. Target Size Study for One-Handed Thumb Use on Small Touchscreen Devices. In *Proc. MobileHCI.* 203–210.

[36] P. Quinn and S. Zhai. 2016. A Cost-Benefit Study of Text Entry Suggestion Interaction. In *Proc. CHI.* 83–88.

[37] P. Quinn and S. Zhai. 2018. Modeling Gesture-Typing Movements. *Hum.-Comput. Interact.* 33, 3 (2018).

[38] K. Reinecke and K. Z. Gajos. 2015. LabintheWild: Conducting Large-Scale Online Experiments With Uncompensated Samples. In *Proc. CSCW.* 1364–1378.

[39] S. Reyal, S. Zhai, and P. O. Kristensson. 2015. Performance and User Experience of Touchscreen and Gesture Keyboards in a Lab Setting and in the Wild. In *Proc. CHI.* 679–688.

[40] K. Tanaka-Ishii, D. Hayakawa, and M. Takeichi. 2003. Acquiring Vocabulary for Predictive Text Entry Through Dynamic Reuse of a Small User Corpus. In *Proc. ACL.* 407–414.

[41] P. D. Varcholik, J. J. LaViola, and C. E. Hughes. 2012. *Establishing a baseline for text entry for a multi-touch virtual keyboard.* Vol. 70. 657–672 pages.

[42] M. Vasiljevas, J. Šalkevičius, T. Gedminas, and R. Damaševičius. 2015. A Prototype Gaze-Controlled Speller for Text Entry. In *Proc. SYSTEM.* 79–83.

[43] K. Vertanen and P. O. Kristensson. 2011. A versatile dataset for text entry evaluations based on genuine mobile emails. In *Proc. MobileHCI.* 295–298.

[44] K. Vertanen and P. O. Kristensson. 2014. Complementing Text Entry Evaluations with a Composition Task. *ACM Trans. Comput.-Hum. Interact.* 21, 2 (2014).

[45] J. O. Wobbrock, A. D. Wilson, and Y. Li. 2007. Gestures without libraries, toolkits or training: A $1 recognizer for user interface prototypes. In *Proc. UIST.* 159–168.

[46] H.-S. Yeo, X.-S. Phang, S. J. Castellucci, P. O. Kristensson, and A. Quigley. 2017. Investigating tilt-based gesture keyboard entry for single-handed text entry on large devices. In *Proc. CHI.* 4194–4202.

[47] F. Zamora-Martínez, V. Frinken, S. España Boquera, M. J. Castro-Bleda, A. Fischer, and H. Bunke. 2014. Neural Network Language Models for Off-Line Handwriting Recognition. *Pattern Recogn.* 47, 4 (2014).

[48] S. Zhai and P. O. Kristensson. 2003. Shorthand Writing on Stylus Keyboard. In *Proc. CHI.* 97–104.

[49] S. Zhai, A. Sue, and J. Accot. 2002. Movement Model, Hits Distribution and Learning in Virtual Keyboarding. In *Proc. CHI.* 17–24.

[50] S. Zhu, J. Zheng, S. Zhai, and X. Bi. 2019. i'sFree: Eyes-Free Gesture Typing via a Touch-Enabled Remote Control. In *Proc. CHI.* 1–12.