ResearchIME: A Mobile Keyboard Application for Studying Free Typing Behaviour in the Wild

Daniel Buschek[1], **Benjamin Bisinger**[1], **Florian Alt**[1,2]
[1]LMU Munich, [2]Munich University of Applied Sciences; Munich, Germany
{daniel.buschek, florian.alt}@ifi.lmu.de, bisinger@cip.ifi.lmu.de

ABSTRACT

We present a data logging concept, tool, and analyses to facilitate studies of everyday mobile touch keyboard use and free typing behaviour: 1) We propose a filtering concept to log typing without recording readable text and assess reactions to filters with a survey (N=349). 2) We release an Android keyboard app and backend that implement this concept. 3) Based on a three-week field study (N=30), we present the first analyses of keyboard use and typing biometrics on such free text typing data in the wild, including speed, postures, apps, auto correction, and word suggestions. We conclude that research on mobile keyboards benefits from observing free typing beyond the lab and discuss ideas for further studies.

ACM Classification Keywords

H.5.2. Information Interfaces and Presentation (e.g. HCI): Input devices and strategies (e.g. mouse, touchscreen)

Author Keywords

Touch keyboard; data logging; typing behaviour; biometrics

INTRODUCTION

We see two main lines of HCI research on mobile touch keyboards today: First, *improving usability and performance* by optimising keyboard layouts [66] with objectives like speed, familiarity, and auto correction [8, 23], also under strong constraints (e.g. only swapping two keys [9]), and by adapting keyboards to individual typists [25], including constraints [30] and context, like hand postures [65]. This addresses the grand goal of fast and error-free (mobile) text entry [41]. Second, *improving privacy and security* by identifying or authenticating [20, 31] users based on behavioural biometrics related to tapping [3, 13] and swiping [12] during text entry, for example to protect data or accounts from unwanted access.

Both keyboard adaptation and biometrics assume that typing varies between users. These research interests motivate studying *individual* behaviour in free text creation in users' everyday lives: Many aspects of individual behaviour may be difficult to observe in lab studies which prescribe fixed settings and behaviours (e.g. hand postures, contexts). The value of taking mobile HCI research beyond the lab has been pointed out both in general [7] and specifically for typing [32]. Recent work on touch keyboards highlights differences in a comparison of lab and field studies [52].

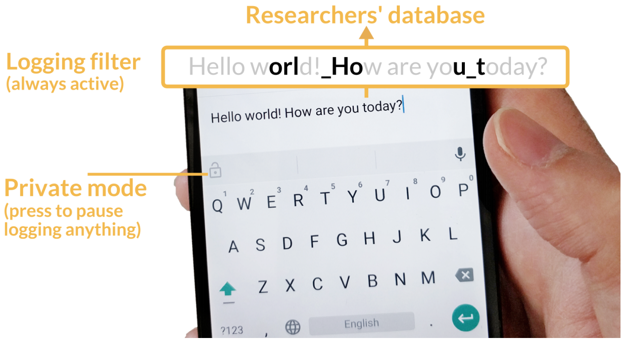

Figure 1. Our *ResearchIME* keyboard app for free text typing studies in the wild, based on Android's open source keyboard. Logged events are sent to the researchers' server. However, data is filtered on the device to avoid logging readable text. The figure shows a random n-gram filter; non-redacted data is highlighted. In addition to the filter, a private mode button allows users to pause logging entirely.

However, studies in the wild so far have used almost exclusively transcription tasks with given text, although text creation tasks are a valuable complementary methodology [61]. This is due to privacy concerns (e.g. [52]), raising the question of how to study typing in natural free text composition without recording readable private conversations.

We address this problem with a novel keyboard app and data filtering concept (Figure 1): It stores all typing events, but omits (*redacts*) the vast majority of touch locations and keys/characters. This promises results of high external and ecological validity. Such data is valuable to inform future keyboard adaptation algorithms and optimisations, as well as biometrics, due to users' varied everyday behaviours and contexts, which contribute to individual keyboard use. Examples for related research questions include:

- *Behaviour*: How are touch keyboards used in everyday typing? In particular, beyond lab measures; e.g. use of "smart" features like suggestions and auto corrections.
- *Biometrics and individualisation*: How well do touch typing biometrics work in the wild? Which features best capture individual behaviour in the wild?

We contribute: 1) A concept for logging typing without readable text, with an online survey (N=349) on different redactions (filters). 2) An open-source implementation as a keyboard app with a first deployment. 3) Analyses and insights into keyboard use and typing biometrics based on 5.9 million keyboard events collected in a three-week field study (N=30).

CHI 2018, April 21–26, 2018, Montreal, QC, Canada.

ACM 978-1-4503-5620-6/18/04...$15.00
https://doi.org/10.1145/3173574.3173829

RELATED WORK

We relate our work to mobile touch keyboard adaptation and optimisation, as well as mobile typing biometrics. In particular, we consider study data and research goals and applications.

Motivating Areas: Optimisation, Adaptation, Biometrics

Many projects *optimise* keyboard layouts [66] with different objectives [8, 23] and constraints [9], and for different mobile form factors and hand postures [50]. Others *adapt* mobile keyboards to typist [25, 30] and context [27] or both [65]. This research typically collects data in the lab to evaluate performance (speed, error rate) of the new design. To compute error rates, researchers have to know the true intended text. This is difficult beyond transcription tasks [26, 46] (judgement by crowd workers is one approach [61]). However, many aspects are insightful to study "in the wild", such as touch distributions [32], auto corrections, word suggestions, and varying contexts. Our concept and tool enable such analyses.

Mobile typing *biometrics* is another area interested in quantifying (individual) typing behaviour, namely to identify or authenticate users [20, 31], also for passwords [3, 13]) and gesture-typing [12]. A 2013 survey [57] found that 73 % of typing biometrics research collected data in one (lab) session. A 2016 follow-up on mobile touch biometrics [58] called for data collection tools. A "roadmap" for mobile biometrics [53] found that comprehensive datasets on keyboards are not available and that the lab influences behaviour. A comparison of implicit authentication methods [38] also concluded that evaluation on real world data is needed, but difficult due to privacy concerns. These conclusions strongly motivate our work.

Studying Typing in the Wild

We argue that the described lines of text entry research could greatly benefit from typing data in the wild, based on three aspects of interest, also motivated by related work:

Assessing real-world variability in keyboard use: A foundation for designing adaptive UIs is the assessment of variability in user behaviour [11]. For mobile touch and typing, several sources and dimensions of behavioural variability have been identified, mostly in the lab, such as: targeting patterns [63], finger placement [34], perceived input point [33], and hand posture [5]. As keyboards become equipped with novel (adaptive) features, studying their natural use in the wild may reveal further user variabilities, such as preferences in input style [52]. For instance, our results indicate individual and context-dependent differences in the use of word suggestions.

Respecting real-world usage contexts: Advanced keyboard adaptation schemes already utilise a variety of context factors, examined in the lab, with adaptations related to key, hand posture, and individual typists [65], as well as body movement [27]. Hence, following related work on typing studies with high external validity [32, 52], it seems adequate to further study (adaptive) keyboard use on data collected in the real world, where contexts vary naturally in everyday interactions. This motivates our work on collecting and analysing such data.

Evaluating real-world performance and user experience: Reyal et al. [52] showed that performance and user experience of mobile touch keyboards vary between lab and field, and that both types of studies should be conducted. They proposed an experience sampling method (ESM) that prompts transcription tasks on users' phones throughout the day. In contrast, we present a logging and filtering concept to observe typing behaviour in users' own natural free text compositions.

In summary, related work shows that there is an opportunity to inform and advance research on mobile touch keyboards and biometrics by studying free text typing in the wild. Existing apps for recording typing data use given text/prompts (e.g. [2, 4, 16]). To the best of our knowledge, we present the first tool and analysis for natural typing, including the use of "smart" keyboard features, like auto correction and word suggestions.

Beyond HCI – Mobile Texting Research

Research on mobile texting in psychology uses a variety of methods, like lab studies with scenarios (e.g. writing an email [21]). In another method, participants submit their last X sent messages [22], or all messages sent within a day [45], after manually removing private information. This does not work well for keyboard research, since typing is hard to "self-report". However, inspired by these methods, our logging concept also gives people the control to manually exclude selected data.

PRIVACY-RESPECTFUL KEYBOARD LOGGING

We use a replacement keyboard app and data filtering on three levels. This section explains how we developed this concept.

I. Supporting Privacy and Trust

Prospective participants may face privacy concerns, depending on their views and relationship to the researchers. Here, we see three main cases: 1) They fully trust the researchers. 2) They trust the researchers, but are concerned that private content might still be looked at when handling the data, even if just by accident. 3) They do not trust the researchers at all. Technical privacy mechanisms seem less of an issue in the first case, and the third case seems difficult to support, since we believe that no one should participate in a study if researchers are not trustworthy. Hence, we focus on the second case.

Scope: Our concept facilitates privacy in the sense that researchers cannot read content when collecting and analysing typing data according to their non-malicious research activities. This aims to help increase participants' trust in the study and supports researchers in conducting privacy-respectful data recording and analyses.

Disclaimer: We do not claim protection against *malicious* attempts to spy on content. While our concept makes this more difficult, a keyboard app alone cannot counter threats like hacking into a database for "de-anonymisation" (i.e. inferring if a user is part of a dataset), which has been shown to be a general threat for large and sparse datasets [48].

II. Requirement Analysis: Data Desired by Related Studies

We conducted a literature search to inform what our logging tool should ideally record. We focussed on work that utilises data on (individual) keyboard use, either to improve usability or privacy and security. Hence, we used the search terms *mobile touch keyboard adaptation/personalisation/optimisation*, and *mobile touch keystroke/typing biometrics* in conference proceedings, the ACM Digital Library, and Google Scholar.

Data (*What?*)	Reveals text?	Data Usage (*How?*)	Study/Application Goal* (*Why?*)
timestamp	no	typing speed	EVL [27, 50, 52]
		inter key times	TXT [27], KBA [65], OPT [23, 50], BIO [13, 31, 67]
		intra key times	TXT [27], BIO [13, 20, 31, 67]
letter/key	yes (if ordered)	entered characters	KBA [6]
		associate data – keys	KBA [6], OPT [23, 50], BIO [20, 31]
		error rate	EVL [30, 50, 52, 62, 65]
touch location	yes (if ordered)	raw values (x, y)	TXT [27], KBA [55], BIO [13, 20]
		distribution per key	TXT [15, 29, 30, 65], BIO [20], ANA [5, 59], KBA [65]
		offset (x, y)	BIO [13, 20, 31]
		key-to-key distance	BIO [13, 37]
		drag (down to up)	TXT [27], BIO [13, 31]
touch pressure	no	raw values	BIO [1, 13, 20, 31, 67]
		in touch model	KBA [62]
touch area	no	raw values	BIO [1, 13, 20, 31, 67], KBA [28]
hand posture	no	independent variable	ANA [5, 59], BIO [13, 67]
		from extra sensors	KBA [17]
		infer from touches	KBA [15, 28, 65]
keyboard size	no	width, height	ANA [54]
inertial sensors	no	raw values, timeseries	TXT [27, 28, 47], BIO [20, 31, 67]
suggestions	yes	independent variable	EVL [51]
auto-correction	yes	on/off, self-reported	ANA [49]
		controlled by behaviour	TXT [62]

* TXT: infer intended text, EVL: evaluate text entry performance, ANA: analyse user behaviour, KBA: adapt/personalise keyboard, OPT: optimise keyboard layout, BIO: authenticate/identify typist

Table 1. Studied keyboard data, based on >70 papers and surveys on mobile (touch) keyboard use, optimisation, adaptation, and biometrics. Columns show *what* data is of interest, whether it could be used to *reconstruct private text* if logged in free composition, *how* the data is considered, and *why* it is useful (with examples, not meant to be exhaustive).

We cannot discuss all reviewed work here, but as an overview, we give starting points for further reading: Teh et al. provide surveys on typing biometrics in general [57] and for mobile touch devices [58]. Overviews also exist for mobile keyboard adaptation/optimisation [39, 46]. Kristensson and Vertanen [41] list recent mobile (touch) keyboards.

We reviewed the work with regard to these questions: 1) Which data is observed, related to keyboard use and typing behaviour? 2) Which behaviour/performance measures are computed? 3) Which models are employed, for analysis and in technical systems? 4) What is the goal of observing this data? This requirement analysis resulted in a list of common data dimensions that should ideally be recorded (Table 1).

III. Logging Strategy: Redacting Text-Revealing Data

Our analysis revealed the sources of concern with regard to private text content: logging keystrokes/characters, touches, as well as word suggestions and auto-corrections.

Several strategies exist: Draffin et al. [20] removed timestamps, shuffling the data. They lost temporal information, which is used in many cases (see Table 1). Kumar et al. [43] logged typing for given tasks in a custom browser; they could only observe this one application. A scheme for desktop typing [19] recorded the top 200 most common English words. It directly aggregated data according to the specific study. This hinders reuse for other analyses. Another desktop logger [24] obfuscated text by replacing characters with *"m"*, yet a lot of work on mobile typing needs to observe behaviour per key.

Promisingly, our analysis showed that the vast majority of evaluations could be extended to free text composition even if we only record keys and touches for *some* keystrokes:

- Keyboard adaptation typically uses touch distributions per key. Most language-aware system parts use short sequences (e.g. *n*-grams) – and language modelling algorithms can be evaluated on text corpora, without keyboard data.
- Research on typing biometrics commonly uses models for key-to-key transitions. Hence, recording keystrokes for selected words or character tuples/triples is sufficient.
- Finally, the (relatively few) studies on word suggestions/corrections are interested in occurrences or availability, which does not require recording the involved words.

Hence, a simple yet viable strategy to support most studies without recording readable text stores all keyboard and typing events, but omits touch locations and keys/characters for the vast majority of these events.

IV. Three Filter Levels: Fixed, Researcher, Participant

We identified three filter levels for the sampling of data that should (not) be redacted, by clustering the concerns and requirements in the related studies.

Filter Level 1 – Fixed Filtering: Some typing data should never be recorded. Filters on this level trigger automatically (e.g. never record data when typing in password fields).

Filter Level 2 – Researcher: A study may only need certain types of data. For example, we could sample based on location, time, text content, word type, app, or random selection.

Filter Level 3 – Participant: This level gives participants direct control over the logging (e.g. with a "recording on/off" button).

V. Choosing Filters and their Parameters

We explored Level 2 filters related to characters, words, and randomness, namely logging: *everything*, only *nouns*, only *verbs*, only *adjectives*, *random words*, the 400 most *common words* in the typist's language (similar to [19]), and *random n-grams*. We chose these filters to cover a range of options which are all rather generally applicable and informative according to our requirements analysis.

Since parameters have the strongest impact on the random schemes, here we focus on discussing those. In particular, to develop a deeper understanding and a more formal perspective of the filter settings, we conducted a "simulated word guessing" analysis: We applied *random n-grams* filters with sample rate p to the words of the enron mobile data [60]. We computed the ratio r of words that would be recognised if a researcher/system still recognises words with edits of up to $x\,\%$ of the word. That is, we use edit distance as a proxy for (human) interpretation. We repeated this 20 times for stability. Figure 2 plots the results as x vs r with lines per filter (i.e. n, p combination). The left plot shows that going beyond $p = 0.1$ drastically increases the ratio of recognised words. By comparison, in the right plot, we see that increasing n has less influence in this analysis. Still, higher n clearly increase the ratio of recognised words as well, as is to be expected.

Based on this analysis, the reviewed research interests, and manual checks looking at filtered text, we chose $n = 3$ and $p = 0.1$. These analyses and our implemented filter also include a minimum gap of one character between loggings to avoid longer n-grams by chance. Table 2 shows examples on text.

VI. Online Survey on Filters

We conducted a survey to assess first reactions to the described Level 2 filters. It was distributed via social networks and a university mailing list, and was completed by 349 people (57 % female) with a mean age of 24 years (range: 17-68 years).

The survey asked people to imagine that they take part in a scientific study on mobile keyboard use, which logs their everyday typing. This scenario stated that data is securely and anonymously stored and analysed and that no passwords are logged. It also mentioned that recording can be turned off temporarily at any time. The survey then presented the filters with the described settings. Each filter was explained with the same seven example sentences in which the characters/words that would be logged by said filter were highlighted in red.

The survey's example sentences were taken from a German university chat corpus[1]. We selected sentences that both cover casual small talk, as well as more sensitive information (e.g. illness of a relative). The purpose of this mix was to provoke ratings for the filters themselves, not for the filters in combination with specific sentences. Participants rated perceived privacy violation on a five-point Likert scale.

Figure 3 shows the results: Logging *everything* was almost unanimously seen as a privacy concern. This demonstrates the need for filters. *Adjectives* and *random n-grams* caused the least critical ratings. A Friedman test with post-hoc analysis showed significant differences between all schemes (all $p < 0.05$), apart from: *random words* vs *common words*, *random n-grams* vs *common words*, and *random n-grams* vs *adjectives*.

VII. Conclusions for our Deployment

For the deployment in this paper, we chose *random n-grams* with $n = 3$, 10 % chance of logging, and a minimum logging gap of one character. This avoids logging whole words and received good subjective ratings in the survey. Table 2 shows an example of using this filter.

The *random n-gram* filter (with small n) is also supported by language properties[2]: 1) identical bi/trigrams occur in many words; 2) most two to three letter words are common ones used by everyone (e.g."the"); 3) due to the exponential distribution of n-grams the vast majority of samples are the language's top ones (e.g. "th"). Thus, similar bi/trigrams are expected across users and an inverse mapping back to words is non-trivial.

This does not mean that nothing could ever be inferred; malicious intent is beyond our scope. However, our hands-on experiences with the chosen filter in our tests and analyses for this paper strongly support the conclusion that no readable text is revealed when handling the filtered data in line with the research interests of the analysed literature.

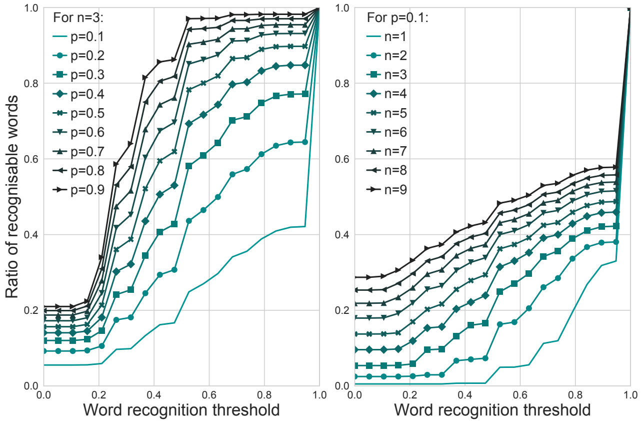

Figure 2. Filter comparison by "simulated word guessing" (see text). Left: different p with $n = 3$. Right: varying n for $p = 0.1$.

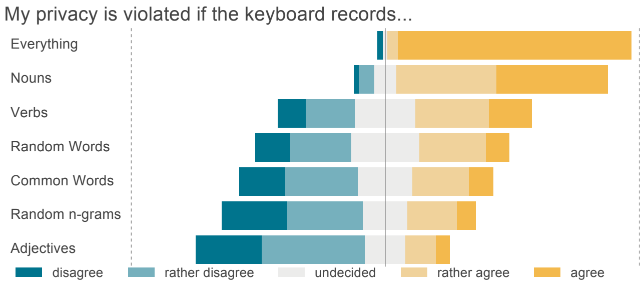

Figure 3. Results from the Likert questions on privacy concerns about different possible Level 2 sampling-based filter schemes ($N = 349$).

The ACM_CHI Conference on_Human Factors in Computing Systems is the premier international_conference_of Human-Computer Interaction. For_first-time attendees,_CHI_is a place_where_researchers and_practitioners gather from_across the world_to discuss the latest_in interactive technology. We are a multicultural community from highly_diverse_backgrounds who together investigate new_and creative ways_for people to_interact. At this year's_CHI - pronounced 'kai' -_the theme will be engage. Our focus will be to engage_with people, to engage with technology, to engage with newcomers,_to engage with world-class research, to engage with your community_of_designers,_researchers,_and practitioners... to engage with_CHI!
_CH; nce; n_H; cto; ern; _co; ce_; ion; _fi; dee; ,_C; _is; _wh; re_; nd_; tit; fro; _ac; d_t; t_i; tec; cul; com; fro; ly_; e_b; nds; _an; _fo; eop; o_i; act; s_C; oun; 'ka; _th; foc; gag; _wi; ech; log; _to; ear; mmu; y_o; _de; s,_; ear; ,_a; tit; ner; ...; h_C
CM_CH; Confe; ence_; Human; ctors; n_Com; ting_; s_is_; he_pr; ier_i; tiona; _conf; _Huma; mpute; racti; ._For; atten; es,_C; _a_pl; e_whe; esear; hers_; tione; ather; _acro; _worl; discu; _the_; t_in_; tive_; echno; gy._W; _a_mu; cultu; al_co; unity; ighly; erse_; ackgr; ds_wh; estig; te_ne; creat; ve_wa; _for_; ople_; _inte; is_ye; HI_-_; ronou; ced_'; _-_th; _them; will_; _enga; ur_fo; o_eng; with_; eople; _to_e; gage_; ith_t; hnolo; o_eng; ge_wi; h_new; mers,; engag; ith_w; d-cla; resea; h,_to; engag; th_yo; commu; y_of_; s,_re; rs,_a; ition; ..._t; ge_wi; CHI!

Table 2. Results of applying our trigram filter with 10% logging probability to the CHI'18 "Welcome from the Chairs" message. Logged spaces are replaced by "_" for visualisation. For comparison, the bottom row shows results for a 5-gram filter with 30% logging probability.

IMPLEMENTATION

Approach: Keyboard App

For security reasons enforced by the OS, the required data (Table 1, e.g. touches) can only be fully accessed by the keyboard. Rooting a device can overcome this, but may restrict or bias the pool of participants. Hence, we decided to replace the keyboard with a custom app. Other work [2] reached the same conclusion. However, their app used a given prompt. This is not suitable to study free everyday typing.

[1] http://www.chatkorpus.tu-dortmund.de/

[2] e.g. www.norvig.com/mayzner.html, *accessed 5th September 2017.*

Keyboard GUI

Our keyboard is built upon Google's Android Open Source Project (AOSP) Keyboard[3]. This allows us to stick as close as possible to the default keyboard that most Android users already know. We did not change or limit any features, but added a small button for the private mode, as explained next.

Private Mode and Other Filters

A private mode button (Figure 4) realises the privacy filter Level 3. No data is recorded while in private mode. The button is always visible while the keyboard is open.

Level 1 filters were implemented based on the text field types provided by Android. In particular, our app automatically switches to private mode for the following fields: password, phone number, person name, postal address, and email address.

As a Level 2 filter, we chose *random n-grams* for our study (with n=3, as in our survey). Our app provides a flexible framework that allows for integration of different (custom) filters.

Data Logging

We record the following data per touch event: timestamp, touch event type (down/move/up), app, hand posture (e.g. "right thumb"), keyboard state (locale, width, height), device orientation (portrait/landscape), touch pressure and size (as reported by the Android API), and other sensor values. Availability of these other sensors and virtual sensors depends on the device – they include: accelerometer, gravity, gyroscope, magnetic field, light, proximity, pressure, relative humidity, and temperature.

We also record "content change" events in the current text entry field, yet we only log the length of the text, not the text itself. Word suggestion picks and auto-corrections are also recorded, without the words, but with basic measures (e.g. word length before/after correction). For the sampled random n-grams (i.e. the "non-redacted" keystrokes), we also record key/character (e.g. "a") and touch location (x, y on keyboard).

Usage and cause of the private mode are also logged for our evaluation. Causes can be automatic activation, for example due to a password field, or manual activation by the user.

Experience Sampling Method (ESM) Module

Besides typing, mobile HCI researchers are often interested in additional context data. This may also include information that is difficult to assess reliably only with device sensors.

To account for such research interests, we developed a simple experience sampling method (ESM) module. It shows a keyboard overlay – only when the keyboard opens to not interrupt typing. Researchers can configure the sampling procedure (e.g. show with random chance or once per hour). In our study, the ESM screen showed up when opening the keyboard, but no more than once every ten minutes. It asked participants to select their current hand posture (Figure 5). Researchers may change this to collect other data. Our pretests showed that it is advisable to not require more than one touch on such screens.

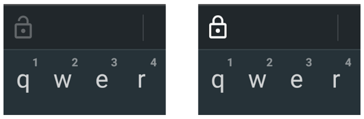

Figure 4. The *private mode* button, added to the left of the word suggestion bar. The symbol shows whether it is inactive (*left*) or activated (*right*). No data is collected in private mode. We chose a symbol instead of text to save space on the keyboard. To ensure that functionality and meaning was clear to all participants, we explained the button and private mode in the initial study meeting.

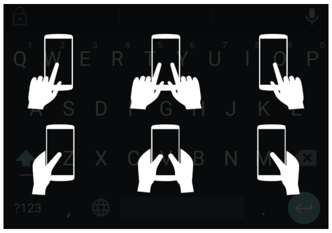

Figure 5. Screenshot (here: for portrait use) of the experience sampling method (ESM) overlay from our study. It asked participants to indicate their current posture by touching the corresponding pictogram, which also made the overlay disappear, revealing the keyboard underneath.

Backend

Our backend is a Java server application, using the Play Framework[4]. The recorded data is transmitted from the keyboard app to this server via HTTPS TLS. The server stores the received data in a MYSQL database. An abstract unique user id identifies data from the same participant. It is generated by the server and sent to the client when connecting for the first time. The backend offers a configuration file to change settings during the study, which are regularly checked by the clients. This allows for custom extensions, for example to change aspects of the keyboard during the study. This is useful for within-subject study designs. The study presented in this paper is explorative and thus did not make use of this feature.

USER STUDY

We conducted an explorative study to analyse everyday keyboard use and test our app as a research tool. With this study we demonstrate what can be computed on this kind of data, "closing the circle" from concept and tool to analyses. Moreover, we contribute new insights into keyboard use and biometrics based on this first free typing data collected with our app.

Apparatus

We employed our app and two questionnaires – one was answered at the start, the other at the end of the study. They assessed demographics, self-perception of behaviour (e.g. hand postures), and study feedback (e.g. private mode, differences to usual keyboard). People used their own smartphones.

Participants

We recruited 30 participants (15 female) via a university newsletter and social media. Their mean age was 24 years (range: 18-33). One was left-handed. 80 % were students, 50 % related to computer science. The vast majority were proficient touch keyboard users. They received a € 15 gift card.

[3] https://android.googlesource.com/platform/packages/inputmethods/LatinIME/, *accessed 5th September 2017.*

[4] https://www.playframework.com/, *accessed 5th September 2017.*

Procedure

We invited participants to our lab to explain the study, the logged data, and the privacy protection scheme. We installed our app on their own devices and configured it to match their usual settings (e.g. haptic feedback, auto correction). We explained the ESM screen and the private mode button. We encouraged them to use it as much as they liked to stop data recording. They also filled in the first questionnaire. After three weeks, we invited participants via email to fill in the final questionnaire and instructed them to uninstall our app.

Limitations

Participants used our keyboard which might have influenced their behaviour, for example due to different visuals and key sizes, yet also different underlying algorithms. However, our app is based on the Android default keyboard; most Android users are likely to be familiar with this keyboard. They also started with Google's Android stock dictionary for auto correction and word suggestions. Future studies should import participants' existing dictionaries.

Hand posture was assessed via ESM and thus might not always be accurate. For example, users might select the posture icon that's easiest to reach. We iterated the design of the selection to reduce effort, yet decided against randomising posture icon order to avoid mistakes due to inconsistency. However, in the post-study questionnaire, all but one person (strongly) agreed with *"I always selected the posture I was actually using"*. Moreover, our data includes a "possibly outdated" flag that indicates if the keyboard has been closed and re-opened after the last ESM answer.

This paper focusses on typing, yet our app is also capable of recording text entry via gestures, which we will analyse in the future. Since gestures produce whole words, filter schemes related to words seem most practical here (e.g. only adjectives).

The online survey mostly reached people from one (Western) country, and study participants were mostly students in tech-related fields. Their privacy concerns and behaviour may not generalise across different cultures and other user groups. However, they are a well-suited sample for a first feasibility study as they are likely to use keyboards a lot.

We cannot entirely rule out that our data contains typing from non-participants due to device sharing, yet all participants reported to have personal "unshared" devices. We also told them to use the private mode if they shared their device temporarily.

DATA SELECTION AND PREPROCESSING

Excluding Typing Breaks for Temporal Analyses

To analyse speed and timing of natural typing, we need to exclude breaks in the typing process that would distort the results (e.g. breaks for thinking, or external interruptions). To achieve this, we define a maximum typing gap time d_t. If a typist takes longer than d_t to press the next key, we assume this to be an interruption of the typing process and do not consider the related keystrokes for computing speed or timing features.

We chose a conservative threshold, setting d_t to four seconds. This is the longest median time *per character*, which occurred for rarely used symbols. Hence, this choice is motivated by excluding breaks, yet retaining delays that we consider part of natural behaviour, such as searching for a rare symbol.

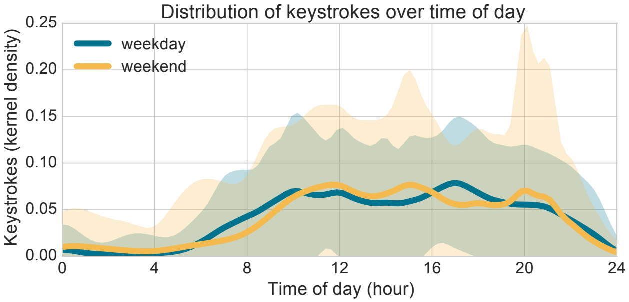

Figure 6. Distribution of keystrokes over the day (shaded $\pm 2\sigma$).

Excluding Other Input Methods for Keystroke Analyses

We logged (anonymous) text entry gestures (see e.g. [42, 52]), yet people rarely used gestures (<4.5 % of entered text). Hence, we focus on typing and exclude gestures. For computing speed, we also exclude events that add more than one character per touch (e.g. auto correction, picking a suggested word).

KEYBOARD USAGE ANALYSES AND RESULTS

We conducted several analyses to cater to a range of reviewed research interests. Thus, we analyse different aspects of keyboard use and biometrics. We analyse behaviour on different levels, including typing activity, speed, postures, orientation and context (apps). We also "zoom in" on aspects that seem particularly interesting to evaluate for free everyday typing instead of given text (e.g. auto correction, deletion, suggestions).

Dataset Overview

We logged 5,930,006 keyboard events, including touch down/move/up, auto corrections, suggestion picks, and text field content changes, plus 27,396,719 sensor readings. There are 963,398 keystrokes (204,685 non-redacted). We analyse non-redacted data whenever keys/characters or touch locations need to be known, otherwise we use all data. The contribution of individuals varied from 1489 keystrokes to 180,948 (median 20,353). The hand posture ESM was answered 7025 times. Inter key intervals had a grand mean of 369 ms (SD 425 ms), excluding breaks as explained in the preprocessing section. The top ten most frequent keys/characters were: space (10.8 %), delete (8.2 %), e (8.1 %), a (5.3 %), i (5.2 %), n (5.1 %), s (4.6 %), r (4.4 %), t (3.9 %), and h (3.6 %).

Typing Activity Over the Day

Figure 6 plots the distribution of keystrokes over the day. On weekdays, peak activities occur before noon and in the afternoon (5 pm). Weekend typing is shifted towards the afternoon, with an evening peak around 8 pm. Rather high standard deviations (shaded) are a result of people's individual behaviour.

Typing Speed

We report speed in words per minute (WPM). A "word" is five characters [52, 64]. Per user, we measure the number of keystrokes T that entered a single character, and the time S for these keystrokes. We thus exclude gestures, suggestions, auto corrections, and breaks. We use S, T to compute WPM with the standard formula (equation 3.1 in [64]). Table 3 shows the results. The mean speed was 32.1 WPM. Comparison between postures is difficult, since not everyone used every posture. However, two thumbs/fingers achieved highest speeds. The correlation of number of keystrokes and mean speed was significant ($r = 0.50, p < 0.01$); people who typed more also typed faster. We found no significant correlation of speed and screen size ($r = -0.08, p > 0.05$).

Hand Postures

We report relative use of postures per person. The most popular postures were *two thumbs* (74.5 %) and *right thumb* (12.7 %). All other postures had a median relative use below 3 %. However, individual participants favoured other postures: For example, one almost exclusively typed with the right index finger, another one used it about 80 % of the time. Posture changes happened for everyone. On average, participants used 4.6 different postures at least once. In our pre-study questionnaire, 28 people said they used *right thumb* (very) often (4 & 5 on 5-point Likert scale). Only 15 said this for *two thumbs*. Thus, people underestimated their relative use of *two thumbs* compared to *right thumb*. One noticed this herself and reported that she was surprised by her amount of typing with two thumbs.

Auto Correction

We recorded 7686 auto corrections (max. per person: 1565). Among its users (N=16), the median number per person/day was 15.3. We found no trend over time. The mean length of corrected words was 5.16 characters prior to correction, 5.24 afterwards. In 7043 cases, correction left the length unchanged, in 546 cases it added at least one character, and 97 corrections removed at least one. The mean Levenshtein distance [44] between entered and corrected word was 1.27.

Word Suggestions

People picked 13,682 word suggestions. The median per person/day was 9.1 among those who used suggestions (N=27). We found no trend over time. The mean ratio was 1.6 %. Thus, people on average picked one suggestion every 63 keystrokes. However, suggestion use was highly individual: The user with the highest ratio had 9.4 %, picking a suggestion every 11 keystrokes. This was also the slowest typist (15.6 WPM). For closer analysis, we recalculated speed for this user *including* words entered via suggestions, which resulted in 34.5 WPM.

We also analysed typing speed calculated *only* on word suggestion picks: We calculated the total number of characters in the suggested words (as the text length T), and the sum of the durations between each suggestion pick and the previous keystroke (as the time taken S). This resulted in a mean speed of 75.6 WPM (95 % CI: 68.9 – 82.2). This provides an estimate of the hypothetical text entry speed that participants in our study could have reached if the displayed word suggestions would have always included their desired next word.

Typing Device Orientation

Only 0.63 % of all keyboard events occurred in landscape mode. 13 people (43 %) always used portrait. One generated 16.2 % of her data in landscape orientation, the second highest participant had 1.7 %, and all others were below 1 %.

WPM	Mean	95% CI	*N*	Keystrokes	
				count	%
all postures	32.1	28.5 – 35.8	30	805,972	100
two thumbs	36.8	31.8 – 41.7	22	119,956	14.9
right thumb	25.3	22.3 – 28.3	28	36,997	4.6
left thumb	27.3	14.2 – 40.5	14	798	0.1
two index fingers	36.7	30.3 – 43.0	18	2498	0.3
right index finger	24.4	20.1 – 28.7	27	15,385	1.9
left index finger	30.8	24.5 – 37.0	20	1696	0.2

Table 3. Typing speeds. Not everyone used every posture (see *N* column). The individual postures do not sum up to 100 %, since the "all postures" row includes events for which the posture is unknown.

Category	Keystrokes		Sugg. ratio (%)	AC ratio (%)
	count	% of data		
Messenger	767129	82.7	**1.6**	**0.9**
Browser	62210	6.7	0.9	0.5
Notes	13677	1.5	1.0	0.4
Email	13465	1.5	1.0	0.6
Social	13324	1.4	0.6	0.1
Shopping	12992	1.4	**1.6**	**0.9**
Communication	12394	1.3	**1.6**	0.4
Maps, Travel & Transport	5984	0.6	0.0	0.0
Search Box	5099	0.5	0	0
Productivity	4780	0.5	0.2	0
Education	4623	0.5	0.9	0
Calendar	3050	0.3	**1.6**	0
Media & Video	1513	0.2	0	0
Other	1315	0.1	0.1	0
Music & Audio	1042	0.1	0.1	0

Table 4. Top 15 app categories ranked by number of keystrokes. Columns show the ratio of suggestion picks and auto corrections to keystrokes, respectively. The highest ratios are highlighted in bold.

Text Deletion

Overall, 8.9 % of non-redacted keystrokes hit the "delete" key (7.6 % for *two thumbs*, 9.0 % for *right index*, and 11.3 % for *right thumb*). Other postures were used too rarely for a meaningful analysis. The ratio of "delete" presses was significantly higher for people who used auto correction ($N = 16$, $M = 0.11$, $SD = 0.04$) than for those who did not ($N = 14$, $M = 0.06$, $SD = 0.02$) (Mann-Whitney U test, $U = 164$, $p < 0.005$).

App Context

Table 4 summarises typing for the top app categories. Broad categories were taken from the Google Play store[5]. They were manually refined to be more meaningful (similar to e.g. [10]).

People used suggestions most often in messengers and other communication apps, as well as calendars and shopping apps. In contrast, few suggestions were picked in maps and transportation apps. This may be explained by more specific vocabulary, not in the dictionary (e.g. location and station names), compared to, for example, a chat context. Additionally, such apps often offer their own GUI elements for suggestions (e.g. based on previously selected locations and routes), which make suggestions shown by the keyboard obsolete.

Likely for similar reasons, auto correction occurred most often in messengers and shopping apps, and (almost) not at all in other categories, again including maps and transportation.

[5] https://play.google.com/store, *accessed 5th September 2017.*

Typing Individuality and Biometrics

Following related work [14], we evaluate individuality of typing behaviour by measuring how well we could distinguish users based on typing characteristics. For this, touch locations were normalised by screen size, as in related work [2, 20].

Keyboard adaptation typically models individual behaviour with Gaussians per key (e.g. x, y touch distribution [6, 29, 30, 65]). We thus evaluate individuality using such a Gaussian *key model* to represent each user. We also evaluate a *transition* variation of this model, which models behaviour per key-to-key transition, as commonly used in keystroke biometrics [57].

For each participant $p \in P$ we train such a model m_p on the training part of p's data. We then feed the testing part of p's data to this model, which predicts probabilities for p (ideally, they would be high). For each other user $q \in P \backslash \{p\}$ we feed the testing part of q's data to the model m_p, which again predicts probabilities for p (ideally, they would be low). We use ten-fold cross validation to train and test on different parts of each participant's data in this procedure. We report mean receiver-operating-characteric area-under-curve (ROC AUC) and equal error rate (EER) of those folds. This is a standard evaluation method for typing biometrics [57]. Hence, these evaluations also demonstrate that such analyses are indeed possible on data collected with our tool and concept.

Table 5 shows results for different features ("drags" are x and y distances between touch down and up; "offsets" describe x, y distances between touch up and key centre). An AUC value of 0.5 is random guessing, 1.0 denotes perfect user separation. The measured values thus show that finger placement is highly user-speficic: touch location and offsets have an AUC above 0.9 and outperformed temporal features.

Moreover, Table 6 shows the results for transition models split by hand posture, limited to the two postures that were used regularly enough by the largest subsets of people for a meaningful analysis. The ranking of features is the same for both postures. Direct posture comparisons are difficult due to different subsets of users, but the results indicate generally higher individuality for *two thumbs* than *right thumb*. This matches related work for typing in the lab [13].

To check if different devices bias the results, we repeated our analyses for the largest subset with the same screen size and aspect ratio (N=8) and with the same device model (N=3). For the first group, the only possible remaining device influence is thus due to the specific device model. However, for our sample (all smartphones), we expect physical screen size to be the most dominant device-related influence on typing behaviour.

For both subsets, we still measured high individuality (same size – AUC: 0.90, EER: 11.43%; same device model – AUC: 0.99, EER: 1.67%; all with transition models and touch locations; compare to Table 5). Thus, following related work [2, 20], normalising spatial features by screen size ensured that device size differences yielded no considerable information. We thus conclude that our analyses show individuality of user behaviour, not device-specific influences.

Feature	key model		transition model	
	ROC AUC	EER (%)	ROC AUC	EER (%)
touch location	0.92	14.85	0.92	15.40
offset x, y	0.91	14.74	0.92	15.48
drag x, y	0.60	42.37	0.60	43.10
intra key time	0.73	30.31	0.70	34.07
inter key time	0.59	42.94	0.64	39.15

Table 5. Individuality of typing features. This table shows ROC AUC scores and equal error rates (EER) per feature, obtained when observing 150 keystrokes, for both 1) unigram models (modelling keys) and 2) bigram models (modelling key transitions).

Feature	right thumb (N=25)		two thumbs (N=20)	
	ROC AUC	EER (%)	ROC AUC	EER (%)
touch location	0.80	26.15	0.93	14.26
offset x, y	0.80	26.04	0.93	14.66
drag x, y	0.60	43.11	0.61	42.11
intra key time	0.68	38.12	0.69	36.18
inter key time	0.56	44.70	0.56	42.76

Table 6. Individuality of typing features for different hand postures (with transition models after 150 keystrokes).

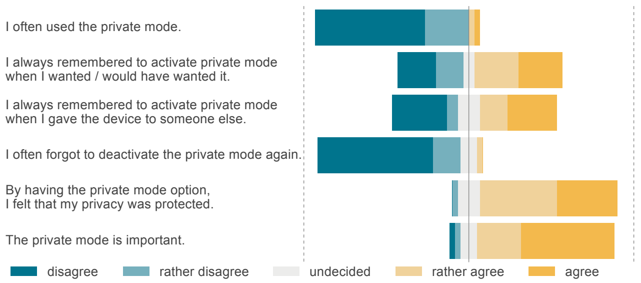

Figure 7. Likert question results on manual use of the private mode.

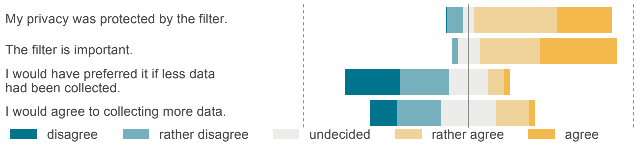

Figure 8. Likert question results on the n-gram filter used in the study.

PERCEPTION OF DATA COLLECTION

Private Mode

Manually activating the private mode (Figure 4) accounts for 12 % of its activations; 88 % were automatic activations, mostly due to password fields (48 %) and phone number fields (20 %). Figure 7 shows that people rarely forgot to deactivate private mode, yet not always activated it if they would have liked to do so. However, the vast majority found it important and felt protected. We asked similar questions about automatic activation: 87 % felt protected by it, and all found it important.

n-gram Filter

The post-study questionnaire explained the n-gram filter again. Figure 8 shows the related questions: Almost everyone felt protected by the filter and rated it as important. 13 % of participants would have preferred it if the app had collected less data, while 23 % would agree to the collection of more data.

DISCUSSION

Respecting Privacy and Related Risks

Our proposed data filters are aimed at facilitating privacy-respectful studies, but not at protecting people from malicious attackers. Besides such attacks, if one tries, sometimes special words might still be guessable in context. For example, observing an *n*-gram like "fac" in a browser app likely indicates "facebook". We argue that intrusiveness in this case is comparable to the common logging of app names in studies.

We rely on other apps' text entry field labels to automatically activate private mode. At least popular apps provide these labels rather consistently, since they are required by accessibility standards (e.g. to read out GUIs for blind users).

People manually used our private mode infrequently, but almost everyone rated it important and felt protected by it. We conclude that participants should be granted such an option to pause data collection from time to time. Our private mode button was always visible, yet some said they did not always remember to use it, which shows that our automatic activation (e.g. for passwords) is also important. The button could be placed more obtrusively, but this might cause distraction or annoyance, hinting at a trade-off to be explored in the future.

Potential Keyboard-specific Influences

As outlined in the study limitations section, replacing the keyboard app may introduce several effects on the resulting behaviour. To minimise those, we configured our app to match participants' usual settings (e.g. haptic feedback on/off) on an individual basis. However, we could not adjust the visuals. For example, different key sizes or border widths compared to a user's usual keyboard might affect finger placement and error rates. Different underlying keyboard algorithms and dictionaries likely also play a role. While algorithms of (closed-source) apps are hard to replicate, we plan to enable visual customisation. Studies could then also replicate the visuals of participants' custom keyboards, not only the Google one.

Many Variables are Measurable on Mostly Redacted Data

To respect privacy and increase trust, we omitted text-revealing information for most logged events. Despite this redaction, many analyses were possible, including typing activity, speed, and biometrics. However, we could not measure error rates. While this is possible for some composition tasks (e.g. judgement of crowd workers [61]), we recorded random trigrams, which a human observer cannot "correct" post-hoc. A future version could assess errors directly on the device with a dictionary (similar to [24]), for example storing error counts. Nevertheless, free text composition enabled us to examine everyday use of suggestions and auto correction. Reflecting on our analyses and participants' feedback, we conclude that the trigram filter was a suitable choice for our study.

Observations on "Smart" Keyboard Features

Only 16 people used auto correction (also on their usual keyboards), yet 27 used word suggestions. Some said that they disliked auto correction since it was often wrong. Our logged data supports this – users of auto correction had significantly higher ratios of delete key presses than those not using it. Others mentioned that they did not often use suggestions, but that they keep them activated since they do not interfere with typing. Hence, we explain the observed difference in popularity with the features' different levels of control: Suggestions are presented for selection by the user, while corrections happen automatically. Weir et al. [62] summarised disadvantages of this loss of control and proposed a method for users to control the "aggressiveness" of correction via touch pressure. Our results with everyday typing highlight the importance of such research to increase utility and acceptance of auto correction.

Comparison to Related Studies

Goel et al. [27] measured 31.1 WPM for their *WalkType* keyboard with transcription in the lab while sitting and walking. Their *ContextType* keyboard [28] had 27.5 WPM, again with lab transcription. Kristensson and Vertanten [41] list a ranking of these and other keyboards. Our results fit into the mid to fast end, yet note that direct comparisons are limited by study differences (e.g. task, phone, keyboard).

To help relate these results, it is also worth noting that we observed rather experienced typists. People might also use different speed-accuracy trade-offs in casual everyday typing (e.g. messaging), compared to what might be perceived as more "formal" settings like a lab study or a field study with dedicated text entry prompt. To investigate this further, more studies of free text composition in the wild are required, which we hope to facilitate with our logging concept and app.

Here, we refine the picture per hand posture: Azenkot and Zhai [5] studied different postures in the lab. Matching our results, they also found that typing with both thumbs was faster than using one. However, their participants were overall faster (e.g. for two thumbs 50 WPM vs our 36.8 WPM), likely due to the different setups (lab vs wild, transcription vs composition). This is supported by Reyal et al. [52], who also reported that typing with both thumbs in the lab was faster compared to the field (35-40 WPM vs 30-34 WPM). Finally, our posture-specific speeds match the order reported by Goel et al. [28] (two thumbs > right thumb > right index finger).

Regarding posture popularity, typing with both thumbs was the most used posture in our study, whereas participants of Azenkot and Zhai [5] reported more balanced use. Fittingly, our participants' self-reports in the initial questionnaire underestimated two thumbs use. This shows the value of assessing such data via our keyboard's ESM module in-situ, not (only) as a questionnaire detached from everyday typing situations.

Natural Keyboard Use is Highly Individual

We found great individuality in keyboard use. For example, one person typed relatively little (and slowly) and used suggestions almost six times as much as others. Another one typed in landscape mode for a sixth of the time, while most others almost never typed in this orientation. While the majority used both thumbs most of the time, some strongly preferred different postures, such as typing with the index finger. Different daily routines resulted in diversity in typing activity throughout the day. Overall, we conclude that capturing behavioural diversity is a strength of studying free text typing in the wild.

Insights into Touch Typing Biometrics

Regarding behaviour features used in keyboard adaptation and typing biometrics, we found that finger placement was more characteristic and consistent than typing rhythm. This difference is in line with results for typing in the lab [13, 35], yet it is even more pronounced for our data. Presumably typing rhythm is less robust under the noise of varying everyday contexts and interruptions, compared to finger placement.

The importance of finding novel biometric features for mobile touch typing was highlighted by related work [18, 57]. For the first time, our results quantify the benefit of such features for everyday typing in the wild with free text composition, including different hand postures and user models. Based on our results, we recommend to use touch locations/offsets for mobile typing biometrics.

Opportunities for Further Studies

As outlined in the related work section, we see several motivating areas of mobile keyboard research. Here we discuss ideas for further studies and application scenarios of our app as a research tool in these areas.

Collecting behaviour data for context-aware keyboards: Following related work on context-aware keyboards (e.g. adapting to walking [27] and hand postures [28, 65]), researchers could use our app to gather typing and sensor data from varying real-world contexts. Our keyboard's ESM screen could be adapted to help with labelling this data (e.g. walking). This data could then inform future context-aware keyboards and their underlying models.

Evaluating keyboard designs in the wild: Looking ahead, researchers could modify the presented keyboard in our app (also see future work section) to study the impact of design choices under different real-world contexts, using our data logging. This could include both simple visual properties (e.g. colours, sizes) as well as functional modifications. For example, based on our results, we could investigate adding a second row of suggestions, as a novel trade-off between screen space and saved typing effort.

Observing keyboard learning behaviour in the wild: Our app could observe how users' behaviour develops as they learn to type with a new layout. This might involve completely new layouts, switching to a different language layout, or layout modifications (e.g. key swaps [9]). The collected data could help to develop new training schemes or further inform models of keyboard learning [36].

Investigating novel keyboard biometrics: Referring back to the example questions in the introduction, our app could further evaluate individual typing behaviour. Our results in this paper contribute insights into the real-world power of spatial and temporal typing features for biometrics. Beyond this, future studies could evaluate if users can be identified based on other keyboard-related behaviour that is particularly interesting to study in the wild. For example, novel keyboard biometrics might recognise users also based on the use of suggestions, the way they delete text, app-specific typing behaviour, or even the filtered logging data itself (e.g. *n*-gram distributions).

CONCLUSION

Mobile touchscreen typing has mostly been studied in the lab with transcription tasks. To facilitate free text entry studies in the wild, we developed a logging concept and keyboard app. A survey (*N*=349) assessed views on related privacy filters. We deployed our app in a three-week field study (*N*=30) and presented the first analyses of keyboard use and typing biometrics in free text composition in the wild, including speed, postures, apps, auto correction, and word suggestions.

Finally, is it worth the effort for researchers to collect typing data from free text composition in everyday life? We believe the answer is yes, it is a valuable additional method. As the HCI community follows diverse research interests with regard to text entry, the study choice should be aligned with those:

The lab transcription task is practical if one is mainly interested in speed and error rates, for example to compare a novel text entry method to a baseline. However, as related work showed [52], these tasks should also be conducted in the field.

On the other hand, our study showed individual differences: For example, a lab study might enforce a fixed (common) hand posture, but we found that some people prefer other postures than the majority. Similar observations hold for device orientation, use of word suggestions, and typing activity throughout the day. Moreover, users' favourite posture in our questionnaire differed from the one they actually reported most often while typing. Our results also suggest that the biometric value of typing behaviour shows in different features in free typing in the wild compared to transcription in the lab.

In conclusion, we see great value in studying unconstrained typing in users' daily lives to capture user-specific behaviour. This fits research agendas on adaptive and personalised keyboards and on typing biometrics. Moreover, such data may help to address the challenge of designing for special user groups and varying contexts of use [39, 40]. By releasing our app and dataset, we hope to support these research endeavours and to encourage further studies of this kind:

http://www.medien.ifi.lmu.de/research-keyboard

FUTURE WORK

We plan to extend the tool further, for example to count whitelisted word occurrences on the device for studies of language use. This could extend recent work on smartphone use in psychology [56]. Moreover, we are working on an API for easy (and remote) configuration of keyboard GUIs (layouts, key sizes, etc.), for example to study alternative layouts and visuals with our app. The concept of sampling random subsequences could also facilitate privacy-respectful collection of other kinds of user data (e.g. location or fitness timeseries).

ACKNOWLEDGEMENTS

We thank Florian Thoma and Timo Erdelt for further development of the *ResearchIME* keyboard app and backend. Work on this project was partially funded by the Bavarian State Ministry of Education, Science and the Arts in the framework of the Centre Digitisation.Bavaria (ZD.B). This research was supported by the Deutsche Forschungsgemeinschaft (DFG), Grant No. AL 1899/2-1.

REFERENCES

1. S. J. Alghamdi and L. A. Elrefaei. 2015. Dynamic User Verification Using Touch Keystroke Based on Medians Vector Proximity. In *Computational Intelligence, Communication Systems and Networks (CICSyN), 2015 7th International Conference on*. 121–126. DOI: http://dx.doi.org/10.1109/CICSyN.2015.31

2. Naif Alotaibi, Emmanuel Pascal Bruno, Michael Coakley, Alexander Gazarov, Stephen Winard, Filip Witkowski, Alecia Copeland, Peter Nebauer, Christopher Keene, and Joshua Williams. 2014. Text Input Biometric System Design for Handheld Devices. In *Proceedings of Student-Faculty Research Day, Pace University*. 1–8. https://pdfs.semanticscholar.org/6bbf/41e3a8b1ceb7ec9cdd4bb2797dd5ebb7a1ff.pdf

3. Margit Antal, László Zsolt Szabó, and Izabella László. 2015. Keystroke Dynamics on Android Platform. *Procedia Technology* 19 (2015), 820–826. DOI: http://dx.doi.org/10.1016/j.protcy.2015.02.118

4. Ahmed Sabbir Arif and Ali Mazalek. 2016. WebTEM: A Web Application to Record Text Entry Metrics. In *Proceedings of the 2016 ACM on Interactive Surfaces and Spaces (ISS '16)*. ACM, New York, NY, USA, 415–420. DOI:http://dx.doi.org/10.1145/2992154.2996791

5. Shiri Azenkot and Shumin Zhai. 2012. Touch Behavior with Different Postures on Soft Smartphone Keyboards. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '12)*. ACM, New York, NY, USA, 251–260. DOI: http://dx.doi.org/10.1145/2371574.2371612

6. Tyler Baldwin and Joyce Chai. 2012. Towards Online Adaptation and Personalization of Key-target Resizing for Mobile Devices. In *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces (IUI '12)*. ACM, New York, NY, USA, 11–20. DOI: http://dx.doi.org/10.1145/2166966.2166969

7. Leon Barnard, Ji Soo Yi, Julie A. Jacko, and Andrew Sears. 2007. Capturing the effects of context on human performance in mobile computing systems. *Personal and Ubiquitous Computing* 11, 2 (2007), 81–96. DOI: http://dx.doi.org/10.1007/s00779-006-0063-x

8. Xiaojun Bi, Tom Ouyang, and Shumin Zhai. 2014. Both Complete and Correct?: Multi-objective Optimization of Touchscreen Keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2297–2306. DOI: http://dx.doi.org/10.1145/2556288.2557414

9. Xiaojun Bi and Shumin Zhai. 2016. IJQwerty: What Difference Does One Key Change Make? Gesture Typing Keyboard Optimization Bounded by One Key Position Change from Qwerty. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 49–58. DOI: http://dx.doi.org/10.1145/2858036.2858421

10. Matthias Böhmer, Brent Hecht, Johannes Schöning, Antonio Krüger, and Gernot Bauer. 2011. Falling Asleep with Angry Birds, Facebook and Kindle: A Large Scale Study on Mobile Application Usage. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11)*. ACM, New York, NY, USA, 47–56. DOI:http://dx.doi.org/10.1145/2037373.2037383

11. D. Browne, P. Totterdell, and M. Norman (Eds.). 1990. *Adaptive User Interfaces*. Academic Press, San Diego, CA, USA.

12. Ulrich Burgbacher and Klaus Hinrichs. 2014. An Implicit Author Verification System for Text Messages Based on Gesture Typing Biometrics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2951–2954. DOI: http://dx.doi.org/10.1145/2556288.2557346

13. Daniel Buschek, Alexander De Luca, and Florian Alt. 2015. Improving Accuracy, Applicability and Usability of Keystroke Biometrics on Mobile Touchscreen Devices. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 1393–1402. DOI: http://dx.doi.org/10.1145/2702123.2702252

14. Daniel Buschek, Alexander De Luca, and Florian Alt. 2016. Evaluating the Influence of Targets and Hand Postures on Touch-based Behavioural Biometrics. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 1349–1361. DOI: http://dx.doi.org/10.1145/2858036.2858165

15. Daniel Buschek, Oliver Schoenleben, and Antti Oulasvirta. 2014. Improving Accuracy in Back-of-device Multitouch Typing: A Clustering-based Approach to Keyboard Updating. In *Proceedings of the 19th International Conference on Intelligent User Interfaces (IUI '14)*. ACM, New York, NY, USA, 57–66. DOI: http://dx.doi.org/10.1145/2557500.2557501

16. Steven J. Castellucci and I. Scott MacKenzie. 2011. Gathering Text Entry Metrics on Android Devices. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, New York, NY, USA, 1507–1512. DOI: http://dx.doi.org/10.1145/1979742.1979799

17. Lung-Pan Cheng, Hsiang-Sheng Liang, Che-Yang Wu, and Mike Y. Chen. 2013. iGrasp: Grasp-based Adaptive Keyboard for Mobile Devices. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 2791–2792. DOI: http://dx.doi.org/10.1145/2468356.2479514

18. Heather Crawford. 2010. Keystroke dynamics: Characteristics and opportunities. In *Eighth Annual International Conference on Privacy Security and Trust (PST)*. 205–212. DOI: http://dx.doi.org/10.1109/PST.2010.5593258

19. Paul S. Dowland and Steven M. Furnell. 2004. *A Long-Term Trial of Keystroke Profiling Using Digraph, Trigraph and Keyword Latencies*. Springer US, Boston, MA, 275–289. DOI: http://dx.doi.org/10.1007/1-4020-8143-X_18

20. Benjamin Draffin, Jiang Zhu, and Joy Zhang. 2014. *KeySens: Passive User Authentication through Micro-behavior Modeling of Soft Keyboard Interaction*. Springer International Publishing, Cham, 184–201. DOI: http://dx.doi.org/10.1007/978-3-319-05452-0_14

21. Michelle Drouin and Claire Davis. 2009. R u txting? Is the Use of Text Speak Hurting Your Literacy? *Journal of Literacy Research* 41, 1 (2009), 46–67. DOI: http://dx.doi.org/10.1080/10862960802695131

22. Michelle Drouin and Brent Driver. 2014. Texting, textese and literacy abilities: A naturalistic study. *Journal of Research in Reading* 37, 3 (2014), 250–267. DOI: http://dx.doi.org/10.1111/j.1467-9817.2012.01532.x

23. Mark Dunlop and John Levine. 2012. Multidimensional Pareto Optimization of Touchscreen Keyboards for Speed, Familiarity and Improved Spell Checking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2669–2678. DOI: http://dx.doi.org/10.1145/2207676.2208659

24. Abigail Evans and Jacob O. Wobbrock. 2012. Taming wild behavior: the input observer for text entry and mouse pointing measures from everyday computer use. In *CHI Conference on Human Factors in Computing Systems, CHI '12, Austin, TX, USA - May 05 - 10, 2012*. 1947–1956. DOI: http://dx.doi.org/10.1145/2207676.2208338

25. Leah Findlater and Jacob Wobbrock. 2012. Personalized Input: Improving Ten-finger Touchscreen Typing Through Automatic Adaptation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 815–824. DOI:http://dx.doi.org/10.1145/2207676.2208520

26. Andrew Fowler, Kurt Partridge, Ciprian Chelba, Xiaojun Bi, Tom Ouyang, and Shumin Zhai. 2015. Effects of Language Modeling and Its Personalization on Touchscreen Typing Performance. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 649–658. DOI: http://dx.doi.org/10.1145/2702123.2702503

27. Mayank Goel, Leah Findlater, and Jacob Wobbrock. 2012. WalkType: Using Accelerometer Data to Accomodate Situational Impairments in Mobile Touch Screen Text Entry. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2687–2696. DOI: http://dx.doi.org/10.1145/2207676.2208662

28. Mayank Goel, Alex Jansen, Travis Mandel, Shwetak N. Patel, and Jacob O. Wobbrock. 2013. ContextType: Using Hand Posture Information to Improve Mobile Touch Screen Text Entry. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2795–2798. DOI: http://dx.doi.org/10.1145/2470654.2481386

29. Joshua Goodman, Gina Venolia, Keith Steury, and Chauncey Parker. 2002. Language Modeling for Soft Keyboards. In *Proceedings of the 7th International Conference on Intelligent User Interfaces (IUI '02)*. ACM, New York, NY, USA, 194–195. DOI: http://dx.doi.org/10.1145/502716.502753

30. Asela Gunawardana, Tim Paek, and Christopher Meek. 2010. Usability Guided Key-target Resizing for Soft Keyboards. In *Proceedings of the 15th International Conference on Intelligent User Interfaces (IUI '10)*. ACM, New York, NY, USA, 111–118. DOI: http://dx.doi.org/10.1145/1719970.1719986

31. Jonathan Gurary, Ye Zhu, Nahed Alnahash, and Huirong Fu. 2016. *Implicit Authentication for Mobile Devices Using Typing Behavior*. Springer International Publishing, Cham, 25–36. DOI: http://dx.doi.org/10.1007/978-3-319-39381-0_3

32. Niels Henze, Enrico Rukzio, and Susanne Boll. 2012. Observational and Experimental Investigation of Typing Behaviour Using Virtual Keyboards for Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2659–2668. DOI: http://dx.doi.org/10.1145/2207676.2208658

33. Christian Holz and Patrick Baudisch. 2010. The Generalized Perceived Input Point Model and How to Double Touch Accuracy by Extracting Fingerprints. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 581–590. DOI: http://dx.doi.org/10.1145/1753326.1753413

34. Christian Holz and Patrick Baudisch. 2011. Understanding Touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 2501–2510. DOI: http://dx.doi.org/10.1145/1978942.1979308

35. Lohit Jain, John V. Monaco, Michael J. Coakley, and Charles C. Tappert. 2014. Passcode Keystroke Biometric Performance on Smartphone Touchscreens is Superior to that on Hardware Keyboards. 2, 4 (2014), 29–33.

36. Jussi P. P. Jokinen, Sayan Sarcar, Antti Oulasvirta, Chaklam Silpasuwanchai, Zhenxin Wang, and Xiangshi Ren. 2017. Modelling Learning of New Keyboard Layouts. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 4203–4215. DOI: http://dx.doi.org/10.1145/3025453.3025580

37. Georgios Kambourakis, Dimitrios Damopoulos, Dimitrios Papamartzivanos, and Emmanouil Pavlidakis. 2016. Introducing touchstroke: keystroke-based authentication system for smartphones. *Security and Communication Networks* 9, 6 (2016), 542–554. DOI: http://dx.doi.org/10.1002/sec.1061

38. Hassan Khan, Aaron Atwater, and Urs Hengartner. 2014. *A Comparative Evaluation of Implicit Authentication Schemes*. Springer International Publishing, Cham, 255–275. DOI: http://dx.doi.org/10.1007/978-3-319-11379-1_13

39. Per Ola Kristensson. 2009. Five challenges for intelligent text entry methods. *AI Magazine* 30, 4 (2009), 85–94. DOI:http://dx.doi.org/10.1609/aimag.v30i4.2269

40. Per Ola Kristensson, Stephen Brewster, James Clawson, Mark Dunlop, Leah Findlater, Poika Isokoski, Benoît Martin, Antti Oulasvirta, Keith Vertanen, and Annalu Waller. 2013. Grand Challenges in Text Entry. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 3315–3318. DOI: http://dx.doi.org/10.1145/2468356.2479675

41. Per Ola Kristensson and Keith Vertanen. 2014. The Inviscid Text Entry Rate and Its Application As a Grand Goal for Mobile Text Entry. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services (MobileHCI '14)*. ACM, New York, NY, USA, 335–338. DOI: http://dx.doi.org/10.1145/2628363.2628405

42. Per-Ola Kristensson and Shumin Zhai. 2004. SHARK2: A Large Vocabulary Shorthand Writing System for Pen-based Computers. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST '04)*. ACM, New York, NY, USA, 43–52. DOI:http://dx.doi.org/10.1145/1029632.1029640

43. Rajesh Kumar, Vir V. Phoba, and Abdul Serwadda. 2016. Continuous Authentication of Smartphone Users by Fusing Typing, Swiping, and Phone Movement Patterns. In *8th IEEE International Conference on Biometrics: Theory, Applications and Systems*. https://www.researchgate.net/profile/Rajesh

44. V. I. Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady* 10 (Feb. 1966), 707–710.

45. Rich Ling and Naomi S. Baron. 2007. Text Messaging and IM. *Journal of Language and Social Psychology* 26, 3 (2007), 291 –298. DOI: http://dx.doi.org/10.1177/0261927X06303480

46. Scott I. Mackenzie and William R Soukoreff. 2002. Text Entry for Mobile Computing: Models and Methods, Theory and Practice. *Human-Computer Interaction* 17, 2-3 (2002), 147–198. DOI: http://dx.doi.org/10.1080/07370024.2002.9667313

47. Josip Musić, Daryl Weir, Roderick Murray-Smith, and Simon Rogers. 2016. Modelling and correcting for the impact of the gait cycle on touch screen typing accuracy. *mUX: The Journal of Mobile User Experience* 5, 1 (2016). DOI:http://dx.doi.org/10.1186/s13678-016-0002-3

48. Arvind Narayanan and Vitaly Shmatikov. 2008. Robust De-anonymization of Large Datasets (How To Break Anonymity of the Netflix Prize Dataset). *IEEE Symposium on Security and Privacy* (2008), 111–125. DOI:http://dx.doi.org/10.1109/SP.2008.33

49. Gene Ouellette and Melissa Michaud. 2016. Generation Text: Relations Among Undergraduates' Use of Text Messaging, Textese, and Language and Literacy Skills. *Canadian Journal of Behavioural Science / Revue canadienne des sciences du comportement* 48, 3 (2016), 217–221. DOI:http://dx.doi.org/10.1037/cbs0000046

50. Antti Oulasvirta, Anna Reichel, Wenbin Li, Yan Zhang, Myroslav Bachynskyi, Keith Vertanen, and Per Ola Kristensson. 2013. Improving Two-thumb Text Entry on Touchscreen Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2765–2774. DOI: http://dx.doi.org/10.1145/2470654.2481383

51. Philip Quinn and Shumin Zhai. 2016. A Cost-Benefit Study of Text Entry Suggestion Interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 83–88. DOI: http://dx.doi.org/10.1145/2858036.2858305

52. Shyam Reyal, Shumin Zhai, and Per Ola Kristensson. 2015. Performance and User Experience of Touchscreen and Gesture Keyboards in a Lab Setting and in the Wild. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 679–688. DOI: http://dx.doi.org/10.1145/2702123.2702597

53. M. Rybnicek, C. Lang-Muhr, and D. Haslinger. 2014. A roadmap to continuous biometric authentication on mobile devices. In *2014 International Wireless Communications and Mobile Computing Conference (IWCMC)*. 122–127. DOI: http://dx.doi.org/10.1109/IWCMC.2014.6906343

54. Eun Jeong Ryu, Minhyeok Kim, Joowoo Lee, Soomin Kim, Jiyoung Hong, Jieun Lee, Minhaeng Cho, and Jinhae Choi. 2016. *Designing Smartphone Keyboard for Elderly Users*. Springer International Publishing, Cham, 439–444. DOI: http://dx.doi.org/10.1007/978-3-319-40548-3_73

55. Christian Sax, Hannes Lau, and Elaine Lawrence. 2011. LiquidKeyboard: An ergonomic, adaptive QWERTY keyboard for touchscreens and surfaces. In *The Fifth International Conference on Digital Society*. 117–122. http://hdl.handle.net/10453/16246

56. Clemens Stachl, Sven Hilbert, Jiew-Quay Au, Daniel Buschek, Alexander De Luca, Bernd Bischl, Heinrich Hussmann, and Markus Bühner. 2013. Personality Traits Predict Smartphone Usage. *European Journal of Personality* (2013). DOI: http://dx.doi.org/10.1002/per.2113

57. Pin Shen Teh, Andrew Beng Jin Teoh, and Shigang Yue. 2013. A Survey of Keystroke Dynamics Biometrics. *The Scientific World Journal* 2013 (2013). DOI: http://dx.doi.org/10.1155/2013/408280

58. Pin Shen Teh, Ning Zhang, Andrew Beng Jin Teoh, and Ke Chen. 2016. A Survey on Touch Dynamics Authentication in Mobile Devices. *Computers & Security* 59, C (2016), 210–235. DOI: http://dx.doi.org/10.1016/j.cose.2016.03.003

59. Christopher Thomas and Brandon Jennings. 2015. Hand Posture's Effect on Touch Screen Text Input Behaviors: A Touch Area Based Study. *CoRR* abs/1504.02134 (2015). http://arxiv.org/abs/1504.02134

60. Keith Vertanen and Per Ola Kristensson. 2011. A versatile dataset for text entry evaluations based on genuine mobile emails. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. ACM Press. DOI:http://dx.doi.org/10.1145/2037373.2037418

61. Keith Vertanen and Per Ola Kristensson. 2014. Complementing text entry evaluations with a composition task. *ACM Transactions on Computer-Human Interaction* 21, 2 (2014), 1–33. DOI: http://dx.doi.org/10.1145/2555691

62. Daryl Weir, Henning Pohl, Simon Rogers, Keith Vertanen, and Per Ola Kristensson. 2014. Uncertain Text Entry on Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2307–2316. DOI: http://dx.doi.org/10.1145/2556288.2557412

63. Daryl Weir, Simon Rogers, Roderick Murray-Smith, and Markus Löchtefeld. 2012. A User-specific Machine Learning Approach for Improving Touch Accuracy on Mobile Devices. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, New York, NY, USA, 465–476. DOI: http://dx.doi.org/10.1145/2380116.2380175

64. Jacob O. Wobbrock. 2010. Measures of Text Entry Performance. In *Text Entry Systems: Mobility, Accessibility, Universality*. Morgan Kaufmann, Chapter 3, 47 – 74.

65. Ying Yin, Tom Yu Ouyang, Kurt Partridge, and Shumin Zhai. 2013. Making Touchscreen Keyboards Adaptive to Keys, Hand Postures, and Individuals: A Hierarchical Spatial Backoff Model Approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2775–2784. DOI: http://dx.doi.org/10.1145/2470654.2481384

66. Shumin Zhai, Michael Hunter, and Barton A. Smith. 2002. Performance Optimization of Virtual Keyboards. *Human-Computer Interaction* 17, 2 (2002), 229–269. DOI:http://dx.doi.org/10.1080/07370024.2002.9667315

67. N. Zheng, K. Bai, H. Huang, and H. Wang. 2014. You Are How You Touch: User Verification on Smartphones via Tapping Behaviors. In *2014 IEEE 22nd International Conference on Network Protocols*. 221–232. DOI: http://dx.doi.org/10.1109/ICNP.2014.43