Quantifying Aversion to Costly Typing Errors in Expert Mobile Text Entry

Nikola Banovic[1], Varun Rao[2,1], Abinaya Saravanan[3,1], Anind K. Dey[1], Jennifer Mankoff[1]
[1]Human-Computer Interaction Institute, CMU [2]PES Institute of Technology [3]Coimbatore Institute Of Technology
Pittsburgh, PA 15213, USA Bangalore, Karnataka, India Coimbatore, Tamil Nadu, India
{nbanovic,jmankoff,anind}@cs.cmu.edu vrao@andrew.cmu.edu abinaya@andrew.cmu.edu

ABSTRACT
Text entry is an increasingly important activity for mobile device users. As a result, increasing text entry speed of expert typists is an important design goal for physical and soft keyboards. Mathematical models that predict text entry speed can help with keyboard design and optimization. Making typing errors when entering text is inevitable. However, current models do not consider how typists themselves reduce the risk of making typing errors (and lower error frequency) by typing more slowly. We demonstrate that users respond to costly typing errors by reducing their typing speed to minimize typing errors. We present a model that estimates the effects of risk aversion to errors on typing speed. We estimate the magnitude of this speed change, and show that disregarding the adjustments to typing speed that expert typists use to reduce typing errors leads to overly optimistic estimates of maximum errorless expert typing speeds.

Author Keywords
Error cost; speed-accuracy tradeoff; typing speed.

ACM Classification Keywords
H.5.2. [Information interfaces and presentation]: User Interfaces – Theory and methods, evaluation/methodology.

INTRODUCTION
Text entry is one of the most important ways in which users interact with their personal computers and mobile devices. Thus, researchers are exploring ways to improve interaction with both physical and virtual keyboards and make them faster. To predict how fast users can enter text, researchers have developed models to estimate the maximum possible expert text entry speed for keyboards [11, 21, 26, 27, 35, 36, 37]. An upper bound represents a best-case scenario for how fast users could type with a given keyboard. However, those models do not consider the typist's willingness to make errors, an unavoidably common aspect of typing [2].

CHI 2017, May 06-11, 2017, Denver, CO, USA
 ISBN 978-1-4503-4655-9/17/05...$15.00
DOI: http://dx.doi.org/10.1145/3025453.3025695

Typing errors on virtual keyboards occur when users press on virtual keys to enter text, because there is uncertainty as to whether they will hit the right target. Keyboard auto-correction features can mitigate the cost of these errors, but not eliminate them. We hypothesize that typists mitigate the cost of errors by adjusting (increasing) their movement time to reduce the likelihood of errors in predictable ways driven by the cost of making errors. This is because users are risk averse when entering text, where risk is defined as the potential of losing something of value—time, in this case.

The relationship between error cost and movement speed has been documented in target-directed pointing [4], and since typing on a virtual keyboard is also a series of target-directed pointing tasks, we hypothesize that the same effect applies to expert typing. However, no existing model of expert typing speed accounts for a typist's change in movement speed to reduce costly errors. Thus, we hypothesize that existing expert typing models are overestimating maximum expert text entry speeds.

Our first contribution is a modified expert text entry speed model, based on [27, 37], that captures pointing speed changes in response to possible costly motor typing errors. We show how to calculate expected error correction times and how movement times are affected by those correction times. We show how both of those values can be used in calculating the resulting upper bound expert single-digit typing speed. We also show how our model can be used to calculate the expected (or actual) mean expert text entry speed—something that existing models cannot do.

Our second contribution is a study that quantifies the impact of costly error correction on movement speed. This study confirms that findings from [4] that show that users reduce their pointing movement time to achieve an optimal speed-accuracy tradeoff during pointing tasks, generalize to text entry pointing tasks. Twenty participants entered all possible pairs of keys using a QWERTY-like mobile virtual keyboard at two cost levels: 1) *riskless*, or no correction cost even when they missed a target, and 2) *costly*, or incurring the cost of pressing an additional "delete" target to "correct" missing a target before moving to the next trial. This study quantifies the impact of the *costly* condition. We found that risk aversion slows even errorless expert text entry speed on average by 7.21% (2.4 words-per-minute).

Our third contribution demonstrates that we can *estimate* expert movement time instead of having to first train expert typists to be able to measure it empirically. We compute two different *riskless* movement time and upper bound text entry speed estimates—one using the traditional Fitts' Law-based model [27, 37] and another using the FFitts-based model for finger input on touchscreen devices [8]. We also compute estimated *costly* movement time and upper bound speeds using our proposed model. We conducted a typing study that tests the external validity of our. We compared model-predicted speeds with empirically measured text entry speeds of the same 20 participants (training set) from the first study and an additional 8 participants (validation set) to ensure the models generalize to people whose pointing data was not used to compute the models. Participants entered phrases from [29] in five 20-minute sessions. Our results show that our model offers a tighter upper bound on expert text entry than both *riskless* models.

We also show that expected mean typing speeds, calculated as a byproduct of our model, offer a reasonable estimate of the study participants' empirically measured typing speeds. Unlike the existing models that estimate the cost of error correction [2], our model accounts for both the cost of error correction and the adjustments to movement speed that are made to avoid errors. This provides a quick way for keyboard designers to accurately estimate user performance on their keyboards when error correction takes time.

Our findings show that not modeling aversion to typing errors leads to overly optimistic estimates of expert text entry speed. We show that such methods need to account for risk aversion in their optimizations to produce a more accurate estimate of movement times between keys. We show that our *costly* model provides a more realistic upper bound on expert text entry speed. Our results also quantify the potential benefit of reducing errors through improved error-correction on expert text entry speeds.

UNDERSTANDING TEXT ENTRY BEHAVIOR

In seeking the fastest and most accurate text entry methods, researchers have developed different approaches and measures to empirically evaluate user performance with keyboards [1, 26, 28, 38, 41]. However, empirical text entry studies take a long time to conduct, and users often do not reach expert level performance during those studies [21, 26]. This is also one of the main barriers for using text entry performance models that are based on the Keystroke-Level Model (KLM) [13, 19] to estimate how long it takes to type individual keys on a keyboard. This is because such models require empirical measures of text entry speed for users of different expertise levels, which is cumbersome for new keyboards where training experts could take a long time.

To estimate the expert speed without running a long text entry study, researchers aim to design predictive models that can estimate the upper bound on users' text entry speed with a given keyboard. This upper bound represents a best-case scenario for how fast users can type on a given keyboard. Soukoreff and MacKenzie [37] developed, and later refined [27], a model of expert text entry speed based on users' pointing behavior. The model was originally intended to evaluate soft keyboards using a stylus, but has since informed related models for other keyboards, such as multi-tap keyboards [36], mini-QWERTY keyboards [11, 27], stroke-based keyboards [20, 35], flick gesture-based keyboards [5], and ambiguous multi-touch keyboards [22].

The main use for those models has been to compare the upper bound text entry speed between different keyboards (*e.g.*, QWERTY-based with other soft keyboard layouts [26, 35, 44]). A second important use is as the objective function in a soft keyboard optimization task [6, 14, 35, 44]. Early optimization methods favored speed over familiar layout (*e.g.*, [44]). However, the lack of adoptability of those early layouts is likely due to users' tendency to stick to familiar layouts [31]. Thus, recent research has explored methods that optimize keyboards for speed and familiarity [6], and speed, familiarity and error correction [14].

Unfortunately, there is often a large gap between the upper bounds produced by such models and empirically measured text entry speeds [21, 26, 27]. One explanation for that gap is that study participants did not reach expert performance. Researchers often estimate when users will reach expert performance by fitting a power function [10] to the empirical data to predict when the power curve will meet the estimated upper bound [26]. However, power curves eventually cross the upper bound and continue to increase. To reconcile this, Isokoski and MacKenzie [21] combined predicted upper bounds and empirically measured speeds to produce a more realistic fit to the data.

Another explanation for the gap is that existing models consider only the linguistic and motor aspects of text entry. For example, the existing models do not consider the cognitive aspects of text entry [33, 43] or users' aversion to typing errors caused by accidentally pressing a wrong key, which are common [2] and costly to correct. Although calculations of empirical text entry speeds include the correction time associated with corrected typing errors [38], the upper bound text entry speed models do not account for other more subtle effects of errors.

When pointing, people change their pointing behavior in response to even small, milliseconds-level time savings [17, 18] and learn to optimize their pointing over the course of performing a pointing task [15]. To minimize their pointing error rate, users adjust their movement time to achieve an optimal speed-accuracy tradeoff [4]. These phenomena could also affect text entry speed. Yet, existing expert text entry models disregard pointing behavior changes that reduce the risk of error. Instead, the motor component of the existing models is based on the traditional Fitts' law [16, 25], which does not take into account pointing behavior changes due to risk aversion. Thus, quantifying users' aversion towards costly errors and exploring the effects of this risk aversion on expert typing speeds will result in a better understanding of users' typing behaviors and more accurate models of expert text entry.

MODELING RISK AVERSION IN EXPERT TEXT ENTRY

In this section, we discuss two different kinds of expert text entry performance models: 1) *riskless* – models that assume that making a typing error is not costly, and 2) *costly* – models that assume that a typing error requires a user to correct the error, which takes time. We define two measures of expert text entry speed: 1) *upper bound text entry speed*, which a best-case scenario top speed a user can achieve with a given keyboard, and 2) *expected mean text entry speed*, which represents the average typing speed with which a user types on a given keyboard.

Riskless Model of Expert Text Entry Performance

The standard, *riskless,* model of expert text entry speed used in most existing work is based on MacKenzie and Soukoreff's [27] newer form of their original expert text entry speed model [37]. In [27], expert text entry speed for a given keyboard is calculated by first identifying a language (*e.g.*, English, French) for which the typing speeds should be calculated and picking a corresponding corpus containing the frequencies of words in that language. Next, the model predicts movement time between each pair of keys on the keyboard.

Traditionally, movement times for keyboards that require the user to press on keys to enter characters are estimated using standard Fitts' law [16, 25], or FFitts for touchscreen mobile devices [8]. Alternatively, these movement times could be empirically collected by measuring the time it takes users to point between every ordered pair of actual keys on the keyboard. For other types of keyboards that do not require users to press on the keys (*e.g.*, gesture-based keyboards) this movement time includes the time to execute a gesture. The model does not make assumptions about how the movement times are calculated, except that no error correction frequency or time is included in the calculations.

The total typing time of a word is simply the sum of the predicted movement time between each of its characters (with a space character starting and ending the word). Thus, the mean time (t_w) to type a word (w) of length n is given by the sum of movement times (MT) between keys that form consecutive character bigrams (including the added spaces) in that word.

$$t_w = \sum_{i=1}^{n-1} MT_{i,i+1} \tag{1}$$

To calculate movement time across a corpus (t_{CORPUS}), the time to enter each word is weighted by its frequency (N_w):

$$t_{CORPUS} = \sum_{w \in CORPUS} N_w \times t_w \tag{2}$$

The total number of all characters in the corpus (n_{CORPUS}) is the sum of the weighted length ($L_w \times N_w$) of all words in the corpus. The mean speed to enter a character on the keyboard in characters per second (v_{CPS}) is given by:

$$v_{CPS} = \frac{n_{CORPUS}}{t_{CORPUS}} \tag{3}$$

This *riskless* model then computes the estimated upper bound text entry speed v_{WPM} in words-per-minute (WPM), assuming 5 characters per word.

$$v_{WPM} = \frac{v_{CPS} \times 60}{5} \tag{4}$$

Costly Model of Expert Text Entry Performance

We refer to the model just described as *riskless* because the time to type each character of each word in a text corpus is calculated with movement times that do not account for adjustments to movement speed that the user makes to avoid errors. Instead it estimates error-free text entry speeds, in a situation where the user is slowing down to avoid errors. Excluding correction time in the calculation of our *costly* model allows comparison with the existing *riskless* model that also predicts error-free typing speed. The *costly* model simply uses the error-free movement time (MT^{costly}) adjusted for the pointing behavior change due to correction time. We therefore rewrite Equation 1:

$$t_w = \sum_{i=1}^{n-1} MT_{i,i+1}^{costly} \tag{5}$$

We expect that the upper bound on expert text entry speed estimated using the *costly* model will be lower than the bound estimated using the *riskless* model. The difference between the upper bounds calculated using the two models is a quantitative measure of risk aversion.

The core of our contribution lies in the calculation of MT^{costly}. To compute *costly* movement times requires knowledge of how users adjust their speed-accuracy tradeoff in response to potentially costly typing errors. MT^{costly} can be empirically collected, by including some correction time when the user misses a key, such as time to press on a dedicated "delete" key before continuing. However, this may require expert typists who have already reached their upper bound typing speeds. Training such typists with new keyboards may be particularly difficult.

Instead, we base our calculations of *costly* movement times on the pointing model by Banovic *et al.* [4], which estimates the optimal movement time (MT) required to minimize the total character completion time (CT) when a typing error results in time-based cost (C). Thus, we use the following equation from [4] to estimate MT^{costly}:

$$CT = \underset{MT^{costly}}{\operatorname{argmin}} (1 - P_e) \cdot MT^{costly} + P_e \cdot (MT^{costly} + C) \tag{6}$$

Equation 6 states that the user will incur only movement time (MT^{costly}) when there is no error, or movement time and the cost of correcting typing errors (C) when the user misses the intended character. The probability of error (P_e) in Equation 6 represents the probability that the user will miss the target when pointing at movement time MT^{costly}.

Equation 6 jointly optimizes MT^{costly} and P_e because they depend on each other. P_e is given by [42]:

$$P_e = 1 - erf\left(\frac{2.066 \cdot \frac{W}{A} \cdot \left(2^{\frac{MT^{costly} - a'}{b'}} - 1\right)}{\sqrt{2}}\right) \quad (7)$$

where *a'* and *b'* values need to be captured over a range of user speed-accuracy strategies [42]. In our *costly* model, we compute *a'* and *b'* from users' observed movement time and error rate (*e.g.*, by minimizing the mean squared error between the observed and model predicted error rate) [4]. Because the typists only incur the cost of error when they notice the error, the probability of error (P_e) in our model corresponds to the errors that the typists notice and correct (*i.e.*, the Corrected Error Rate [38]).

We rewrite Equation 6 in its shortened form:

$$CT = \underset{MT^{costly}}{\operatorname{argmin}} MT^{costly} + P_e \cdot C \quad (8)$$

Computing C: Cost of Typing Error

To optimize the completion time function above requires the cost of missing each character (C). We modify the original cost calculations from [4] to correspond to target pointing when typing. To correct a missed character, the user has to: 1) delete the wrong character by moving to and pressing the delete key (CT_{del}) and 2) attempt to press the correct character again ($CT_{correct}$). Thus we rewrite Equation 8 as:

$$CT = \underset{MT}{\operatorname{argmin}} MT^{costly} + P_e \cdot (CT_{del} + CT_{correct}) \quad (9)$$

A delete key may also be missed in error, and the user must again point at delete key until a successful hit. Thus, we can compute this time as:

$$CT_{del} = MT_{del} + P_{e_{del}} \cdot C_{del} \quad (10)$$

where MT_{del} is the time to move to and press the delete key and C_{del} is the cost of another mistake. We compute MT_{del} using FFitts' [8] using the distance (A) from the center of the intended key to the delete key, and width of the delete key (W). We use the center of the intended key as the start point because it approximates the mean 2D position of all errors around the intended key. When the user misses the delete key, the distance, and thus movement time to the delete key, gets smaller on each attempt. Note that every time the user misses the delete key, the user has to press one more time on the delete key to correct the new erroneous character they typed (when some other key was pressed instead). We denote the cost of pressing the delete key again as C_{tap}. This can be estimated as the intercept computed using FFitts [8]. Thus CT_{del} is bounded by:

$$CT_{del} \leq \lim_{n \to \infty} \sum_{i=0}^{n} MT_{del} + P_{e_{del}}^{\,i} \cdot \left(MT_{del} + C_{tap}\right) \quad (11)$$

Equation 11 then reduces to:

$$CT_{del} \leq 2 \cdot MT_{del} + C_{tap} \quad (12)$$

We use the right hand side of Equation 12 as the upper bound on the cost of attempting to delete a wrong character. Finally, we compute $CT_{correct}$ also using Equation 9. Thus, we write the final equation for estimating *costly* movement time as:

$$CT = \underset{MT^{costly}}{\operatorname{argmin}} MT^{costly} + P_e \cdot \left(2 \cdot MT_{del} + C_{tap} + CT_{correct}\right) \quad (13)$$

The total cost of error in Equation 13 represents how much the time it takes to correct an error contributes to the expected character completion time. For simplicity, our model assumes that the user will correct only errors that the user notices as they occur.

We then use Equation 13 for each pair of keys on the keyboard to calculate the *costly* movement times that we use in Equation 5 to estimate the time it takes to type each word in the corpus.

Modeling Mean Expert Typing Speeds

An advantage of our *costly* model is the ability to estimate the expected mean expert text entry speed, which includes both character entry time and possible correction time when the user makes a typing error. The *riskless* model does not consider error correction time and cannot be used for such calculations. To estimate the expected mean expert speed, calculations should include the time users spend correcting errors because the empirical text entry speed includes the time to correct errors [38].

A byproduct of optimizing Equation 13 is that we can calculate the expected character completion time (CT), which includes both the time to correctly type the character when the user is pressing the keys at movement time MT^{costly} and the time to correct the error should the user miss the character. Thus, we can calculate the total expected completion time for each word in the corpus:

$$t_w = \sum_{i=1}^{n-1} CT_{i,i+1} \quad (14)$$

By using Equation 14 instead of Equation 1, we can estimate mean expert text entry speed over a corpus as described by Equations 2 through 4.

This calculation of expected mean typing speeds is a simplification of the typing process. It does not model all of the possible correction strategies that users employ when typing [2]. It also does not consider all cognitive aspects of text entry [43]. Our goal is to explore how a single parameter affects the estimate of mean typing speeds. Having fewer parameters enables us to estimate speeds quickly when the estimate is reasonably accurate. We later compare the expected mean text entry speed calculated using our model with empirically measured speeds.

STUDY OF RISK AVERSION IN EXPERT TEXT ENTRY

Our first study tests the hypothesis that the risk of making an error causes typists to reduce their pointing movement time and error rate. Past work showed this effect in target directed pointing [4]. Since mobile touchscreen text entry is a form of target directed pointing, we would expect it to generalize. In this study, we 1) show that the effect is present in expert mobile touchscreen typing behavior, and 2) empirically quantify the size of the effect of costly errors on movement speed and typing speed. As described below, this study uses empirically measured movement times instead of Fitts' law estimated movement times to minimize any inaccuracies in model calculations [37].

Study Design

To collect information about every possible targeting action during typing, we asked participants to move back and forth three times between every possible pair of keys (including the same key twice). To increase the reliability of our measures, participants repeated each pointing three times before the next began. The order of key pairs was random.

Our goal was to elicit behavior similar to expert typing performance. Thus, all keys in the study interface except the current starting key and target were blank to minimize the need for visual search (Figure 1). To further increase expertise, the study included practice trials involving 100 ordered pairs of keys. Before the practice trials, we instructed participants to point at targets as fast and as accurately as possible.

The study used a within subject design with all participants performing the task in two conditions: 1) *riskless* condition, in which participants continued to the next trial even when they selected the wrong key, and 2) *costly* condition, in which participants had to press an additional "delete" key every time they pressed a wrong key, and then press on the correct key, to simulate correction time (Figure 1.B and Figure 1.C). When participants missed a key in the *costly* condition, the experimental interface marked the missed key in red and the "delete" key as current target in white. The order of conditions was counterbalanced.

Participants completed two blocks of practice trials for each condition in alternating order (4 blocks total). In each practice block, participants pressed 100 different ordered pairs of keys (10 keys × 10 keys × 3 trials per pair = 300 trials per block). Participants completed a total of 1,200 training trials. The practice trials allowed participants to experience both conditions and learn to optimally balance their speed-accuracy tradeoffs and approximate expert behavior before moving on to the test trials.

After the practice trials, participants completed 2,187 test trials in which they moved between each pair of keys on the keyboard 3 times in each direction, a total of 27 keys × 27 keys × 3. This resulted in total of 4,374 test trials. The test trials in each condition were split into three blocks, where each block contained 729 different ordered pairs of keys, to allow participants to take breaks between blocks. There were no repeated key pairs between test blocks.

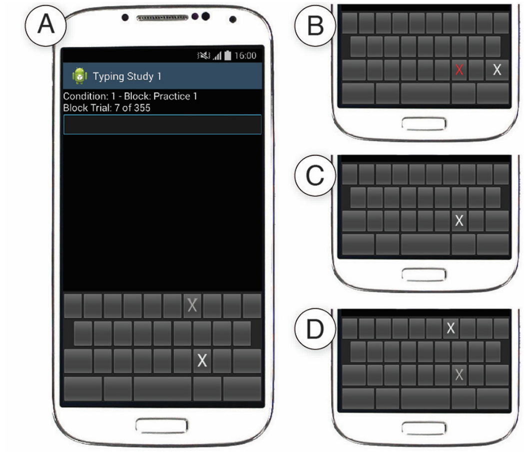

Figure 1. Experimental interface for the study of expert text entry performance: A) interface showing study progress (top) and the blank mini soft keyboard with the current pair of keys marked with white (current target) and light gray (next target) X marks; B) in the *costly* condition, a missed key is marked in red and the "delete" key is marked as current target in white; C) after "delete" is pressed, the missed key from A is marked again as the current target; D) pressing on the correct key switches the color of keys. Note that in the *riskless* condition, the interface skips steps B and C even if the participant misses the next key in step A.

Participants performed the experimental tasks on a Samsung GALAXY S4 smartphone, with a 5-inch Full HD Super AMOLED (1920 × 1080; 441 ppi) display, running Android 4.2.2 (Jelly Bean). All non-essential sensors and services were disabled on the phone.

Key sizes were 306 × 135 pixels (~0.7 × 0.3 in) for the space key, 153 × 135 pixels (~0.35 × 0.3 in) for the delete key, and 90 × 135 pixels (~0.2 × 0.3 in) for all other keys participants pressed on. The keyboard also had five distractor keys that were not used in the study, but are present on most mini-QWERTY soft keyboards (*e.g.*, shift key, enter key). The minimum distance between the centers of two keys was 108 pixels (~0.25 in), and the maximum distance was 1170 pixels (~2.65 in).

Task and Procedures

Participants were briefed on the task and signed a consent form at the study start. Participants then completed a questionnaire asking them about their demographics and their mobile text entry use. Participants then performed the pointing tasks. They pressed on targets with the thumb of their dominant hand. We picked single thumb input because it is a frequently used way to enter text on mobile devices [3]. This also simplified the analysis and comparison of the two models without loss of generality. Participants then completed the 1,200 training trials (in four blocks) and the 4,374 test trials (in six blocks). Participants were allowed to rest between blocks of trials, but not within blocks.

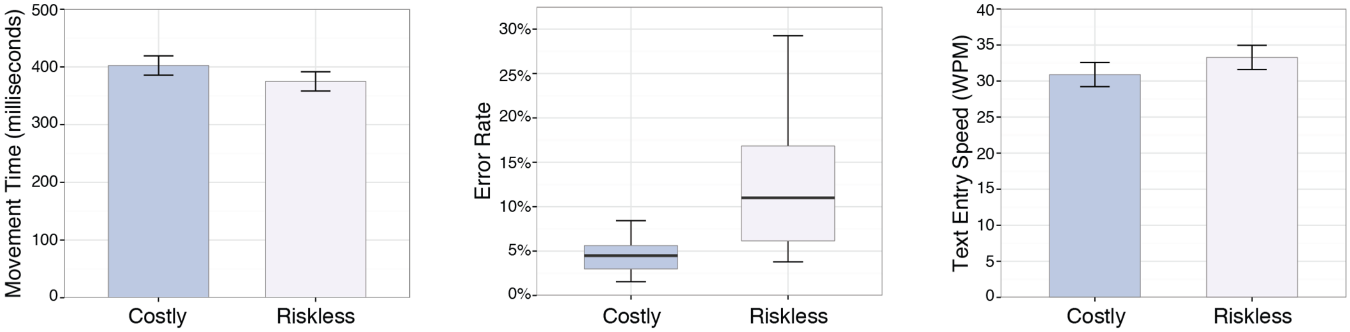

Figure 2. (left) Mean error-free movement times; (middle) median error rates; (right) mean text entry speed calculated using the *costly* model and error-free movement times in the *costly* condition and the *riskless* model and error-free movement times in the *riskless* condition. Error bars in bar charts represent 95% confidence intervals, and minimum and maximum values in the boxplot.

Twenty participants (12 male, 8 female), all right-handed, ages between 18 and 45 (median=24) took part in the study. One participant had more than six months and others had more than a year experience entering text on QWERTY-like mobile touchscreen soft keyboards. All participants entered text using those keyboards daily. Participants were recruited through flyers and an online participant pool. They were compensated $15 for taking part in the study. The study took approximately an hour to complete.

Measures

We measured participants' movement time between each ordered pair of keys on the keyboard. Since we had 3 samples for each ordered pair in each condition, we calculated the median movement time for each ordered pair of keys for each condition. When a mistake was made (a target was missed), we excluded it from the movement time calculation. When participants made mistakes in all 3 trials for an ordered pair, we imputed the data using the median movement time of the reversed ordered pair. When participants made mistakes in the reverse direction as well, we imputed the data using the median movement times across all trials in that condition. We imputed 0.6% of the values in the data.

We calculated the upper bound expert typing speed for each participant for each condition using their empirical movement times and error rates to analyze the effect of risk aversion on typing speeds. We use the *error-free median movement times* from the two conditions to predict the upper bound on expert text entry speed for each participant using Equation 4. The movement time in the *costly* condition was used for MT^{costly} as described in Equation 5 and the movement time (MT) in the *riskless* condition in Equation 1. We used the frequency of words from the Corpus of Contemporary American English (COCA) [12] to calculate upper bounds for both conditions.

We compare the data from the two conditions using paired t-tests. We use Morey's method [30] to compute accurate standard deviation and confidence intervals around means for repeated measures data. We ensured data normality using Shapiro-Wilk tests. When the normality assumption was violated, (*e.g.*, for error rate data), we used non-parametric Wilcoxon tests.

Results

The primary purpose of this study was to measure the degree of risk aversion of expert typists in terms of the effect of risk on error-free movement time. We summarize the results in Figure 2.

The degree of risk incurred in the *costly* condition, measured as the mean correction time it took participants to press the delete key and then press the original missed key again, was 555.65 milliseconds (SD=146.04). The pointing error rate indicates that the participants were risk averse given these costs. Participants were more than twice as accurate in the *costly* condition than in the *riskless* condition (W=0, Z=3.92, $p<.0001$, r=.88). The median error rate, shown in Figure 2 was 4.48% in the *costly* condition compared to 10.99% in the *riskless* condition.

To reduce the error rate, the participants had to reduce their movement speed. Mean movement time in the *costly* condition was 402.31 milliseconds (SD=35.62) compared to 374.97 milliseconds (SD=35.62) in the *riskless* condition, a difference of 27.34 milliseconds. We calculated mean movement time here as the mean of the median movement times for all pairs of keys. This difference was significant ($t_{(19)}=2.43$, $p=.0253$, Cohen's d=.54). This validates that the results in [4] generalize to raw typing movement time.

Next, we examine what the impact of these movement time changes are on the overall predicted errorless typing speed (using the model in Equation 4, which calculates WPM over a corpus and takes into account word frequencies). Movement times resulted in a 2.4 WPM difference between upper bound speeds estimated using the two models. The mean upper bound speed estimated using the *costly* movement times was 30.88 WPM (SD=3.58) compared to 33.26 WPM (SD=3.58) estimated using the *riskless* movement times (Figure 2), a significant difference ($t_{(19)}=2.10$, $p=.0496$, Cohen's d=.47).

Discussion

The results of our study confirmed our intuition that upper bound typing speeds will differ when the risk of making a costly error is present. Our study confirms that participants adjust their pointing behavior by increasing their error-free movement time and reducing their overall error rate when missing a target in text entry tasks is costly. Even when no typing errors occur, this effect significantly reduces upper bound expert typing speed on a typical mobile soft keyboard from 33.26 to 30.88 WPM, a 7.21% difference in speed. This difference of 2.4 WPM quantifies the risk aversion of expert typists when character correction time on average takes 555.65 milliseconds.

The *riskless* condition might appear as an unrealistic manipulation where participants can disregard pointing accuracy; yet, existing published models assume exactly this condition (*e.g.,* [8, 27, 37]). The impact of removing error correction from studies is that movement times and error rates both increase (as was seen in our study). In some cases, study participants reach error rates as high as 60% [8]—error rates that are not practical when entering text.

Although our model remedies the inaccuracies that arise when the impact of errors on error-free movement time is ignored, our study of risk aversion requires an onerous amount of data collection to determine costly movement times. Also, the movement time data we collected is specific to a particular QWERTY-based keyboard, and may not generalize to other keyboards. Thus, we next compute the upper bounds on expert text entry using movement times and error rates computed using target directed pointing models, which generalize across keyboards.

CALCULATING EXPERT TEXT ENTRY SPEED

The empirical measures we used in the previous study demonstrated a measurable difference in movement speed driven by the cost of error correction. However, measuring those variables requires that a keyboard with a known layout already exists, making them less useful for tasks such as optimization-based keyboard design. Thus, in this section we *estimate* the upper bounds and the expected mean text entry speed. We demonstrate that traditional approaches found in the literature for estimating movement time (Fitts' Law [25] and FFitts [8]) approximate empirically computed *riskless* upper bound typing speed from our first study well. Next, we show that our *costly* model closely approximates empirically computed *costly* upper bound speeds from our first study.

We first computed two different *riskless* movement times for each pair of keyboard keys: MT^{Fitts} based on traditional Fitts' Law [25] and MT^{FFitts} based on FFitts' [8]. We did this to compare our *costly* movement times with two existing, accepted, but different ways to compute *riskless* movement times. For both *riskless* models we used a and b values computed using data from studies conducted in [8]. Note that participants in [8] pointed using their index finger. The difference between one thumb and index finger text entry, even if it exists, is very small [3], and thus we expect the pointing parameters to be very similar.

We used Equation 13 to compute movement times (MT^{costly}) between each pair of keys for our *costly* model. We also used FFitts [8] a and b values to compute MT_{del} and CT_{tap} in our calculations of *costly* movement time. To compute P_e, we computed a' and b' values that minimize the mean squared error between the typing error rates we collected in our first study and the error rates we obtain from Equation 7. We used our calculations of MT^{costly} to compute the expected total character completion time CT^{costly}.

Once we computed MT^{Fitts}, MT^{FFitts}, and MT^{costly}, we calculate estimated upper bound typing speeds over the COCA corpus [12], in words per minute, using the same approach as our empirical study (Equations 1 through 4). Similarly, we used CT^{costly} to calculate the expected mean expert speeds. Thus, we had four separate calculated speed measures: v^{Fitts} and v^{FFitts}, which are *riskless* estimates of typing speeds over the corpus, and v^{costly} and v^{CT}, which are our costly model based estimates of the upper bound and expected mean expert speeds over the corpus, respectively.

Using these calculated measures, we found that the two *riskless* upper bound models (v^{Fitts} and v^{FFitts}) were even higher than the empirically calculated *riskless* upper bound from our first study (33.26 WPM), with v^{FFitts} being closer to that estimate (33.59 WPM) than v^{Fitts} (38.43 WPM). Our v^{costly} was 30.30 WPM, which closely matched the *costly* upper bound of 30.88 WPM we computed in our first study from empirically measured *costly* movement times. Our v^{CT}, the mean expert typing speed, was 27.85 WPM.

These results suggest that v^{costly} and v^{FFitst} best match the empirically estimated *costly* and *riskless* upper bounds respectively. FFitts-based v^{FFitts} improves on the traditional v^{Fitts} upper bound by 4.84 WPM by capturing the difficulty of finger touch screen text entry. Our model v^{costly} further improves on the FFitts-based upper bound by 3.29 WPM by mathematically modeling participants' risk aversion as we showed in our first study.

Although we have demonstrated that we can accurately predict the upper bound speeds, this result is at some level theoretical: our calculation is based on estimated movement times derived from target directed pointing models under *riskless* and *costly* conditions, and knowledge of word frequency derived from a text corpus [12]. Our next study empirically validates these results by comparing model-predicted upper bound text entry speeds to expert text entry performance in a real typing task, which always includes some cost when the users makes a typing error.

EMPIRICAL MEASURE OF RISK IN TEXT ENTRY

Our next study was designed to test the validity of our models on previously unseen participants in a real typing task. Thus, our study had two participant groups (*training* and *validation*). The real typing task helps us to test how well our model estimates more realistic typing behavior in *costly* conditions. The *validation* group helps us to test how well the model-estimated expert text entry speeds generalize to other people.

We hypothesize that both the traditional *riskless* and our *costly* models provide upper bounds on empirical expert text entry speed, but that the *costly* model is a closer match to actual expert text entry behavior. Additionally, we hypothesize that the expected mean text entry speeds calculated using our model predict empirically measured speeds well. Finally, we hypothesize that our estimates will be equally accurate for both the *training* group of participants (on whom we trained our probability of error function for our *costy* model) and the *validation* group of participants. Testing on real text entry behavior and previously unseen participants increases our confidence in the external validity of the models.

Study Design

We split participants into two groups: 1) *Training Group*, which consisted of the 20 participants whose data was used to train parts of our *costly* model, and 2) *Validation Group*, which included 8 additional participants whose data we did not train on. The additional 8 participants (6 male, 2 female) were ages between 21 and 34 (median=25). One participant had more than 3 months and all other participants more than a year experience entering text on mini-QWERTY mobile touchscreen soft keyboards, and entered text using these keyboards daily.

Both groups of participants transcribed phrases from [29] using the same QWERTY-like keyboard layout from the first study with key labels added (Figure 3). We used a well-established format for running multi-session empirical text entry studies [26]. Participants transcribed phrases in 5 separate 20-minute sessions. Any two sessions were at least two hours apart and at most two days apart.

Participants transcribed phrases in blocks of ten phrases, and were allowed to rest between blocks, but not within blocks. The first block of every session was a warm-up block and was not used in our analysis. The same phrase never repeated within a block, but repeats were allowed between blocks. In total, each participant completed 50 warm-up phrases (one block each session) and the median number of transcribed testing phrases per participant was 352 phrases (min=260, max=501) in the training group and 350 phrases (min=160, max=414) in the validation group. The total number of transcribed testing phrases during the study for all participants and all sessions in the training group was 7,493 phrases, and 2,712 phrases in the validation group.

Participants performed the experimental tasks on the same Samsung GALAXY S4 smartphone, with a 5 inch Full HD Super AMOLED (1920 × 1080; 441 ppi) display, running Android 4.2.2 (Jelly Bean) as in the first study. All non-essential sensors and services were disabled on the phone. All measures were calculated using the StreamAnalyzer software [39].

Figure 3. A) Experimental interface for the empirical evaluation of the models: B) study progress, current phrase to transcribe, and a text area for transcribed characters, and C) mini QWERTY soft keyboard with blank distractor keys.

Task and Procedures

Participants arrived at our testing lab and were briefed on the task. Those participants that did not participate in the first study signed a consent form and filled out the same questionnaire as in the first study. We asked participants to transcribe phrases as fast and as accurately as possible using the thumb on their dominant hand. Participants were asked to imagine they are writing a text message to a friend and match that level of accuracy. The participants then proceeded to transcribe phrases for 20 minutes. After 20 minutes elapsed, the investigator asked participants to stop typing, which ended the session. We collected data over 5 sessions to allow the participants to adjust to the mini-QWERTY keyboard used in our study. Participants were recruited through flyers and an online participant pool. They were compensated $75 for taking part in the study.

Measures

We measured participants' mean typing speed in words-per-minute (WPM), averaged across all sessions for each group of participants (giving us empirically measured speeds v^{train} and v^{val}). We also measured participants' *mean total error rate*, as the mean of the sum of uncorrected and corrected errors per session [38], to show that even expert typists make errors and choose to correct some of them. We removed any transcribed phrases that had more than a 25% uncorrected error rate because those were likely accidental mistakes where the participants pressed on the enter key too soon or attempted to transcribe phrases too fast and the transcript was not legible. We removed 0.3% of all transcribed phrases due to high error rates.

To check for learning [10], we performed the standard practice of fitting a power function to the mean text entry speed in each session [26]. Participants demonstrated little learning over the 5 sessions, as illustrated in Figure 4, and as expected since we recruited experts. This was confirmed by fitting their data to a power law curve [26] for the training and validation group: $y = 28.26 \cdot x^{0.0207}$ ($R^2 =$

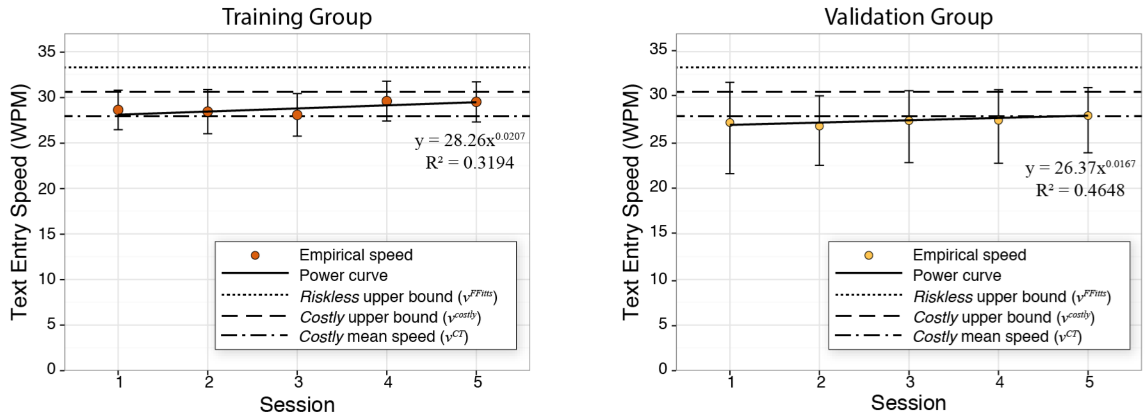

Figure 4. Mean empirically measured text entry speed for the training group (left) and validation group (right) and upper bound speeds estimated using the *costly* model and the *riskless* model based on FFitts. The error bars represent 95% confidence intervals.

0.32) and $y = 26.37 \cdot x^{0.0167} (R^2 = 0.46)$, respectively. Low R^2 values point to the lack of a learning effect rather than other error variance (*e.g.*, like poor measures, distracted or unmotivated participants). We designed our study to minimize any poor measures and any distractions to the participants.

We also verify that even experienced typists make errors. In the training group, the median uncorrected error rate was 0.69% and the median corrected error rate was 6.02%. In the validation group, the median uncorrected error rate was 0.60% and corrected error rate was 6.88%. Some of these errors could be due to accidental errors and slips [32]. However, it is also likely that participants optimized their movement time, which still results in some pointing errors, to minimize their overall typing time [4]. This is similar to the pointing behavior we measured in the first study.

Results

In this section we show that the upper bound typing speeds estimated using the two models represent top typing speeds for the given keyboard. Additionally, we show that the expected mean text entry speed calculated using our model predicts the empirically measured speeds well.

Validating Upper Bound Expert Speed Calculations

To test the validity of our model estimated *costly* upper bound speed, v^{costly}, we compare it to v^{train} and v^{val}, the empirically measured mean typing speeds of each group of study participants. If our upper bound is closer to v^{train} and v^{val} than the two *riskless* model upper bounds, then we can say that it represents a tighter and more accurate upper bound expert text entry speed. Figure 4 shows the comparison between the model estimated upper bound speeds and the empirically measured typing speeds in the two groups. The mean v^{train} is 28.83 WPM (SD=4.77) and mean v^{val} is 26.80 WPM (SD=4.66). The model estimated expected mean speed, v^{CT} is 27.85 WPM and is within the 95% confidence interval of both v^{train} and v^{val} (Figure 4).

Risk Aversion Magnitude

The only difference between calculations of our *costly* upper bound speed (v^{costly}) and *riskless* upper bound speed (v^{FFitts}) is movement time that takes into account the cost of error. Thus, we can conclude that the difference between the two upper bounds measures the magnitude of the participants' risk aversion to costly typing errors. The difference between v^{costly} and v^{train} was 1.47 WPM compared to 4.76 WPM and 9.60 WPM difference for the v^{FFitts} and v^{Fitts} respectively. In other words, 76.40% of the difference between the best, standard *riskless* model predicted speed and the empirical speeds in the training condition is due to participants' risk aversion (Figure 4). Similarly, in the validation group, the difference between v^{costly} and v^{val} was 3.50 WPM compared to 6.79 WPM and 11.63 WPM difference between v^{val} and v^{FFitts} and v^{Fitts} respectively. Risk aversion contributed 65.98% of the difference between the best *riskless* model upper bound and the empirically measured speed (Figure 4).

Discussion

Our model provides a framework to estimate the upper bound expert typing speed (v^{costly}), as well as the expected mean expert typing speed (v^{CT}), which includes pointing movement speed, corrected error rate, and the correction time. We provided empirical evidence to show that v^{costly} represents an accurate upper bound: the mean empirical speeds do not exceed the estimated upper bounds and the power curves we fitted to the empirical speeds are not increasing quickly on the observed interval. The expected mean speed calculated using our model (v^{CT}) was within the 95% confidence interval of the mean empirically measured speeds in our training and validation sets. Our validation set results show that our *costly* model will accurately estimate the upper bound (v^{costly}) and expected mean speed (v^{CT}) for experts whose data was not used to train the model.

Our results suggest that most of the difference between the upper bound calculated using the existing *riskless* model (v^{FFitts}) and the empirical speeds (v^{train} and v^{val}) is due to typists' aversion to costly errors. Around 70% of that difference is due to participants' risk averse behavior of slowing down their maximum typing speed to avoid errors. As a result, the errorless text entry speed calculated using our *costly* model (v^{costly}) results in a more accurate prediction of maximum expert text entry speed than the errorless speeds estimated using the FFitts-based *riskless* model (v^{FFitts}), when typing errors have a cost.

Although both *riskless* and *costly* upper bound speed models assume error-free text entry performance, experts still make errors as suggested by empirical error rates we measured in our study. The error correction time included in our expected mean text entry speeds estimate (v^{CT}) accounts for the difference between our v^{costly} upper bound and the empirically measured expert text entry speeds (v^{train} and v^{val}). Accidental errors and slips are unavoidable [32], so experts tend to point at keys with movement times that optimize their overall performance, but still results in occasional errors [4]. Also, our simplified model assumes that typists correct their typing errors as the errors occur. However, users at times miss the errors and have to fix the errors later. To support such interactions, future work should extend our expected mean typing speed (v^{CT}) calculations to account for the probability that the user will notice errors after they continue typing after an error [2].

Part of the difference between the estimated upper bound (v^{costly}) and the empirical speeds (v^{train} and v^{val}) could also be the time it takes participants to switch their attention between the keyboard and the transcribed text to ensure there are no errors. This is important for models that estimate text entry speed on touch screen surfaces because even expert users might have to glance at the keyboard to avoid drifting from keys in the absence of haptic feedback [24]. Our proposed model does not account for this. One way to account for this difference could be to estimate the gaze time using the Keystroke-Level Model (KLM) for touchscreen devices [33], and include this time in the model. However, this requires further investigation.

We significantly improved the accuracy of upper bound expert text entry speed calculations. The difference between our upper bound (v^{costly}) and the best *riskless* upper bound v^{FFitts} was 3.29 WPM. This shows that such existing models are overly optimistic and may offer unrealistic bounds when designing and evaluating new keyboards. This could in particular be the case with novel keyboards for wearable devices that have a potentially high chance of making costly typing errors due to small screen and key sizes. In such cases, our *costly* speed models will offer a more realistic evaluation of such new keyboards.

CONCLUSION AND FUTURE WORK

In this paper, we have presented a modification to existing models of expert text entry speed that enables us to quantify people's aversion towards making typing errors and more accurately predict their errorless typing speed than existing models. We empirically demonstrated that this phenomenon exists in a study of typing-like target-directed motions with penalties on erroneous presses. We then showed that we can accurately predict upper bound expert text entry speed (v^{costly}) and expected mean expert typing speed (v^{CT}), which both include estimated movement times, which typists modified in response to costly errors. We validated our speed calculations in an empirical study of typing behavior that naturally includes a penalty of error correction.

Our model calculations do not require empirical data once initial FFitts parameters *a* and *b* and error probability function parameters *a'* and *b'* have been obtained. Thus, it can be used for tasks such as keyboard layout optimization in which a rapid calculation of an objective function is needed. In addition, it can easily estimate the impact of error correction, or be used to compare different options for error correction. In future work, we would like to validate that our model generalizes to other keyboards by comparing it to empirical measures of actual speed on those keyboards. Existing keyboard layout optimization methods (*e.g.*, [6, 14]) use the original Fitts' law digraph model to calculate movement times between keys [26, 37]. In addition to using cost-adjusted movement times [4], those methods should explore ways to include the expected correction time for missing individual keys as a parameter when minimizing text entry time on keyboards.

Future research should also explore ways to adapt the *costly* model to estimate expert speeds for different hand postures (*e.g.*, two-thumb text entry), modalities other than finger input (*e.g.*, stylus), and other keyboards (*e.g.*, ambiguous keys [22] and gesture-based keyboards [5]). Future research should also extend the *costly* model to keyboards with common auto-correction and auto-completion features. We focus on transcription, but future research should explore how to adapt our model to composition tasks [39].

Our findings underline the need to explore other variables that might have an effect on expert text entry (*e.g.*, gaze time, cognitive load). Our modification to the existing *riskless* models for calculating typing movement time is just an example of how people's risk aversion to costly errors impacts models based on target-directed pointing. Our findings provide motivation to validate other behavior models that involve pointing, but do not take into account the effects of the cost of error when missing the target.

Finally, our findings have implications beyond calculating expert text entry speeds. As research moves away from optimizing keyboard layouts for speed towards optimizing automated error correction and word completion [7, 9], our work underlines the importance of lowering the cost of error to increase the overall text entry speed. For example, our results imply that good error correction methods can improve errorless text entry. The expert speeds predicted using the existing model could be used to predict text entry when experts are using an ideal autocorrect feature. However, future work should explore how the cost of errors impacts text entry performance when using the existing, less-than-ideal auto-correction algorithms.

ACKNOWLEDGEMENTS

This research was supported by NSERC post-graduate fellowship (PGSD3-438429-2013).

REFERENCES

1. Ahmed Sabbir Arif and Wolfgang Stuerzlinger. 2009. Analysis of text entry performance metrics. In Science and Technology for Humanity (TIC-STH), 2009 IEEE Toronto International Conference, 100-105. http://dx.doi.org/10.1109/TIC-STH.2009.5444533
2. Ahmed Sabbir Arif and Wolfgang Stuerzlinger. 2010. Predicting the cost of error correction in character-based text entry technologies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '10). ACM, New York, NY, USA, 5-14. http://doi.acm.org/10.1145/1753326.1753329
3. 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. http://doi.acm.org/10.1145/2371574.2371612
4. Nikola Banovic, Tovi Grossman, and George Fitzmaurice. 2013. The effect of time-based cost of error in target-directed pointing tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13). ACM, New York, NY, USA, 1373-1382. http://doi.acm.org/10.1145/2470654.2466181
5. Nikola Banovic, Koji Yatani, and Khai N. Truong. 2013. Escape-Keyboard: A Sight-Free One-Handed Text Entry Method for Mobile Touch-screen Devices. *Int. J. Mob. Hum. Comput. Interact.*5, 3, 42-61. http://dx.doi.org/10.4018/jmhci.2013070103
6. Xiaojun Bi, Barton A. Smith, and Shumin Zhai. 2010. Quasi-qwerty soft keyboard optimization. In*Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '10). ACM, New York, NY, USA, 283-286. http://doi.acm.org/10.1145/1753326.1753367
7. Xiaojun Bi, Shiri Azenkot, Kurt Partridge, and Shumin Zhai. 2013. Octopus: evaluating touchscreen keyboard correction and recognition algorithms via “remulation”. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13). ACM, New York, NY, USA, 543-552. http://doi.acm.org/10.1145/2470654.2470732
8. Xiaojun Bi, Yang Li, and Shumin Zhai. 2013. FFitts law: modeling finger touch with fitts' law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13). ACM, New York, NY, USA, 1363-1372. http://doi.acm.org/10.1145/2470654.2466180
9. Xiaojun Bi, Tom Ouyang, and Shumin Zhai. 2014. Both complete and correct?: multi-objective optimization of touchscreen keyboard. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems* (CHI '14). ACM, New York, NY, USA, 2297-2306. http://doi.acm.org/10.1145/2556288.2557414
10. Stuart K. Card, Allen Newell, and Thomas P. Moran. 1983. *The Psychology of Human-Computer Interaction*. L. Erlbaum Assoc. Inc., Hillsdale, NJ.
11. Edward Clarkson, Kent Lyons, James Clawson, and Thad Starner. 2007. Revisiting and validating a model of two-thumb text entry. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '07). ACM, New York, NY, 163-166. http://doi.acm.org/10.1145/1240624.1240650.
12. Mark Davies. 2012. *Word frequency data from the Corpus of Contemporary American English (COCA)*. Retrieved May 15, 2012 from http://www.wordfrequency.info
13. Mark D. Dunlop and Andrew Crossan. 2000. Predictive text entry methods for mobile phones. *Personal Technologies*, *4*(2-3), 134-143. http://dx.doi.org/10.1007/BF01324120
14. 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. http://doi.acm.org/10.1145/2207676.2208659
15. Digby Elliott, Steven Hansen, Jocelyn Mendoza, and Luc Tremblay. 2004. Learning to optimize speed, accuracy, and energy expenditure: a framework for understanding speed-accuracy relations in goal-directed aiming. *J. Motor Behav.*, *36*(3), 339-351. http://dx.doi.org/10.3200/JMBR.36.3.339-351
16. Paul M. Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47, 6, Jun 1954, 381-391. http://dx.doi.org/10.1037/h0055392
17. Wayne D. Gray and Deborah A. Boehm-Davis. 2000. Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *Journal of Experimental Psychology: Applied, 6(4)*, Dec 2000, 322-335. http://dx.doi.org/10.1037/1076-898X.6.4.322
18. Wayne D. Gray, Chris R. Sims, Wai-Tat Fu, and Michael J. Schoelles. 2006. The soft constraints hypothesis: A rational analysis approach to resource allocation for interactive behavior. *Psychological Review, 113(3)*, Jul 2006, 461-482. http://dx.doi.org/10.1037/0033-295X.113.3.461
19. How Yijue and Min-Yen Kan. 2005. Optimizing predictive text entry for short message service on mobile phones. In *Proceedings of 11th International*

Conference on Human-Computer Interaction (HCII 05), Las Vegas, United States.

20. Poika Isokoski. 2001. Model for unistroke writing time. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '01). ACM, New York, NY, USA, 357-364. http://doi.acm.org/10.1145/365024.365299
21. Poika Isokoski and I. Scott MacKenzie. 2003. Combined model for text entry rate development. In*CHI '03 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '03). ACM, New York, NY, USA, 752-753. http://doi.acm.org/10.1145/765891.765970
22. Bonnie E. John. 1996. TYPIST: a theory of performance in skilled typing. *Hum.-Comput. Interact.* 11, 4 (December 1996), 321-355. http://dx.doi.org/10.1207/s15327051hci1104_2
23. Frank Chun Yat Li, Richard T. Guy, Koji Yatani, and Khai N. Truong. 2011. The 1line keyboard: a QWERTY layout in a single line. In *Proceedings of the 24th annual ACM symposium on User interface software and technology* (UIST '11). ACM, New York, NY, USA, 461-470. http://doi.acm.org/10.1145/2047196.2047257
24. Frank Chun Yat Li, Leah Findlater, and Khai N. Truong. 2013. Effects of hand drift while typing on touchscreens. In *Proceedings of Graphics Interface 2013* (GI '13). Canadian Information Processing Society, Toronto, Ont., Canada, 95-98.
25. I. Scott MacKenzie. 1992. Fitts' law as a research and design tool in human-computer interaction.*Hum.-Comput. Interact.* 7, 1 (March 1992), 91-139. http://dx.doi.org/10.1207/s15327051hci0701_3
26. I. Scott MacKenzie and Shawn X. Zhang. 1999. The design and evaluation of a high-performance soft keyboard. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*(CHI '99). ACM, New York, NY, USA, 25-31. http://doi.acm.org/10.1145/302979.302983
27. I. Scott MacKenzie and R. William Soukoreff. 2002. A model of two-thumb text entry. *SPACE* 67 18-43.
28. I. Scott MacKenzie and R. William Soukoreff. 2002. Text entry for mobile computing: models and methods, theory and practice. *Human–Computer Interaction*, *17*(2-3), 147-198. http://www.tandfonline.com/doi/abs/10.1080/07370024.2002.9667313
29. I. Scott MacKenzie and R. William Soukoreff. 2003. Phrase sets for evaluating text entry techniques. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '03). ACM, New York, NY, USA, 754-755. http://doi.acm.org/10.1145/765891.765971
30. Richard D. Morey. 2008. Confidence intervals from normalized data: A correction to Cousineau (2005). *Reason*, *4*(2), 61-64.
31. Donald A. Norman and Diane Fisher. 1982. Why alphabetic keyboards are not easy to use: Keyboard layout doesn't much matter. *Human Factors: The J. of the Human Factors and Ergonomics Society*, *24*(5), 509-519. http://doi.org/10.1177/001872088202400502
32. Donald A. Norman. 2002. *The Design of Everyday Things*. Basic Books, Inc., New York, NY, USA.
33. Kenton P. O'Hara and Stephen J. Payne. 1999. Planning and the user interface: The effects of lockout time and error recovery cost. *International Journal of Human-Computer Studies*, *50*(1), 41-59.
34. Andriy Pavlovych and Wolfgang Stuerzlinger. 2004. Model for non-expert text entry speed on 12-button phone keypads. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '04). ACM, New York, NY, USA, 351-358. http://doi.acm.org/10.1145/985692.985737
35. Jochen Rick. 2010. Performance optimizations of virtual keyboards for stroke-based text entry on a touch-based tabletop. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology* (UIST '10). ACM, New York, NY, USA, 77-86. http://doi.acm.org/10.1145/1866029.1866043
36. Miika Silfverberg, I. Scott MacKenzie, and Panu Korhonen. 2000. Predicting text entry speed on mobile phones. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*(CHI '00). ACM, New York, NY, USA, 9-16. http://doi.acm.org/10.1145/332040.332044
37. R. William Soukoreff and I. Scott MacKenzie. 1995. Theoretical upper and lower bounds on typing speed using a stylus and a soft keyboard. *Behaviour & Information Technology*, *14*(6), 370-379. http://dx.doi.org/10.1080/01449299508914656
38. R. William Soukoreff and I. Scott MacKenzie. 2003. Metrics for text entry research: an evaluation of MSD and KSPC, and a new unified error metric. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '03). ACM, New York, NY, USA, 113-120. http://doi.acm.org/10.1145/642611.642632
39. Keith Vertanen and Per Ola Kristensson. 2014. Complementing text entry evaluations with a composition task. *ACM Trans. Comput.-Hum. Interact.* 21, 2, Article 8 (February 2014), 33 pages. http://dx.doi.org/10.1145/2555691
40. Jacob O. Wobbrock and Brad A. Myers. 2006. Analyzing the input stream for character- level errors in unconstrained text entry evaluations. *ACM Trans.*

Comput.-Hum. Interact. 13, 4 (December 2006), 458-489. http://doi.acm.org/10.1145/1188816.1188819

41. Jacob O. Wobbrock. 2007. Measures of text entry performance. In *Text Entry Systems: Mobility, Accessibility, Universality,* I. S. MacKenzie and K. Tanaka-Ishii (eds.). Morgan Kaufmann, 47-74.

42. Jacob O. Wobbrock, Edward Cutrell, Susumu Harada, and I. Scott MacKenzie. 2008. An error model for pointing based on Fitts' law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '08). ACM, New York, NY, USA, 1613-1622. http://doi.acm.org/10.1145/1357054.1357306

43. Motonori Yamaguchi, Matthew J. C. Crump, and Gordon D. Logan. 2013. Speed–accuracy trade-off in skilled typewriting: decomposing the contributions of hierarchical control loops. *Journal of Experimental Psychology: Human Perception and Performance, 39, 3*, 678-699. http://doi.org/10.1037/a0030512

44. Shumin Zhai, Michael Hunter, M., and Barton A. Smith. 2002. Performance optimization of virtual keyboards. *Human–Computer Interaction*, *17*(2-3), 229-268. http://www.tandfonline.com/doi/abs/10.1080/07370024.2002.9667315