Analysis of Text Entry Performance Metrics

Ahmed Sabbir Arif, Wolfgang Stuerzlinger
Dept. of Computer Science & Engineering
York University
Toronto, Canada
{asarif, wolfgang}@cse.yorku.ca

***Abstract*—Researchers have proposed many text entry systems to enable users to perform this frequent task as quickly and precise as possible. Unfortunately the reported data varies widely and it is difficult to extract meaningful average entry speeds and error rates from this body of work. In this article we collect data from well-designed and well-reported experiments for the most important text entry methods, including those for handheld devices. Our survey results show that thumb keyboard is the fastest text entry method after the standard QWERTY keyboard, and that Twiddler is fastest amongst non-QWERTY methods. Moreover, we survey how text entry errors were handled in these studies. Finally, we conducted a user study to detect which effect different error-handling methodologies have on text entry performance metrics. Our study results show that the way human errors are handled has indeed a significant effect on all frequently used error metrics.**

Keywords-text entry; text entry metrics; error correction; human factors; input devices, strategies and methods

I. INTRODUCTION

Researchers have proposed many systems to enable users to perform text entry as quickly and precise as possible. Recently, most of this work has focused on text entry on handheld devices, since the use of phones is not limited to making calls anymore. These devices are now used for several everyday tasks, like scheduling, text messaging, live chat, task listing, imaging, etc. A major portion of these tasks involves entering text. Unfortunately, experimental data on text entry performance reported in the literature varies widely due to the use of different performance metrics and experimental designs. Hence, it is difficult to compare studies or to extract meaningful average text entry speeds and error rates from this body of work. This makes it hard for designers and researchers to use and apply these results and works against the synthesis of a larger picture.

We begin this article with an introduction of the most common performance metrics used in text entry studies. Then we present data collected from well-reported experiments for seven important text entry methods: two full-length QWERTY keyboards (standard and projection), two reduced-size QWERTY keyboards (thumb and soft); two keypads (Twiddler and standard 12-key keypad), and one stylus based text entry method (Graffiti). We then attempt to harmonize the reported data so that it becomes easier to compare methods and to extract meaningful averages.

Most text entry experiments are conducted with one of three error correction conditions: *none*, *recommended*, and *forced* error correction. In the *none* condition participants are not allowed to correct errors, in the *recommended* condition correction of errors is recommended if and as participants identify them, and with the *forced* condition participants are forced to correct each error. Toward this end, we present a study that investigates if these conditions have a noticeable effect on text entry performance metrics, as this constitutes a good step towards making it easier to compare studies.

II. TEXT ENTRY PERFORMANCE METRICS

In the field of text entry, several metrics are used to characterize a method's performance [1]. Here, we discuss the most common performance metrics employed. In the literature different notations and terms are used to describe various concepts. For better understanding and to avoid confusion we discuss all metrics using the notations formerly introduced by Soukoreff and MacKenzie [2]:

- *Presented Text (P)* is what participants had to enter, and $|P|$ is the length of P.
- *Transcribed Text (T)* is the final text entered by the participant, and $|T|$ is the length of T.
- *Input Stream (IS)* is the text that contains all keystrokes performed while entering the presented text, and $|IS|$ is the length of IS.
- *Correct (C)* is the number of correct characters in the transcribed text.
- *Incorrect Not Fixed (INF)* is the number of unnoticed errors (incorrect characters) in the transcribed text.
- *Fixes (F)* are keystrokes in the input stream, which are edit functions (backspace, delete, cursor movement, etc), modifier keys (shift, alt, control, etc.), or navigation keys (left, right, mouse click, etc.).
- *Incorrect Fixed (IF)* keystrokes are those in the input stream that are not editing keys (F), but which do not appear in the final transcribed text result.
- *Minimum String Distance (MSD)* is the minimum number of operations needed to transform T into P, where the operations are insertion, deletion, or substitution of a single character.

Soukoreff and MacKenzie also introduced a simplification, $INF = MSD(P, T)$, and $C = max(|P|, |T|) - MSD(P, T)$, which consider only the size of P and T [2, 4].

A. Entry Rates

Calculating the text entry rate for various input methods is usually straightforward and simple. The Words per Minute (*WPM*) metric is the most frequently used empirical measure of text entry performance [3]. A few other metrics exist, but are rarely used: Gestures per Second (*GPS*), Adjusted Words per Minute (*AdjWPM*), and Keystrokes per Second (*KSPS*).

1) Words per Minute (WPM)

Word per Minute (*WPM*) measures the time it takes to produce certain number of words. *WPM* does not consider the number of keystrokes nor the gestures made during the text entry but only the length of the transcribed text. *WPM* is defined as:

$$WPM = \frac{|T|-1}{S} \times 60 \times \frac{1}{5}. \quad (1)$$

Here, S is time in seconds measured from the first key press to the last, including backspaces and other edit and modifier keys. The constant 60 is the number of seconds per minute, and the factor of one fifth accounts for the average length of a word in characters including spaces, numbers, and other printable characters [3]. Note that S is measured from the entry of the very first character to the last, which means that the entry of the first character is never timed, which is the motivation for the "–1" in the numerator of (1). While this assures accuracy, other researchers sometimes omit this factor.

B. Error Rates

Unlike entry rates, measuring the error rate is more complex. There are many error rate metrics that are used and none of them are perfect as they all face difficulties distinguishing errors corrected during text entry, and those that remain after (i.e. uncorrected errors). Here we discuss the five most frequently used error metrics:

1) Error Rate (ER)

Error Rate (*ER*) is traditionally calculated as the ratio of the total number of incorrect characters in the transcribed text to the length of the transcribed text:

$$ER = \frac{INF}{|T|} \times 100\%. \quad (2)$$

2) Minimum String Distance Error Rate (MSD ER)

The Minimum String Distance Error Rate (*MSD ER*) metric was introduced based on the application of the Levenshtein string distance statistic [4] to the problem of matching (incorrect) input to the target text. The algorithm yields the minimum distance between two strings (*MSD*) defined in terms of edit operations like *insertion*, *deletion,* and *subtraction* of a single character. The idea is to find the smallest number of operations to transform the transcribed text to match the presented text, and then to calculate the ratio of that number to the larger of the length of the presented and transcribed text:

$$MSD\ ER = \frac{MSD(P,T)}{Max(|P|,|T|)} \times 100\%. \quad (3)$$

Here, $MSD(P, T)$ is the minimum string distance between the presented and transcribed text. Later an improved version of the *MSD ER* was proposed [2], which uses the ASCII representation of the differences between the presented and transcribed text to address the disparity in lengths.

3) Keystroke per Character (KSPC)

Keystroke per Character (*KSPC*) is simply the ratio of the length of the input stream to the length of the transcribed text:

$$KSPC = \frac{|IS|}{|T|}. \quad (4)$$

4) Erroneous Keystroke Error Rate (EKS ER)

Erroneous Keystroke Error Rate (*EKS ER*) measures the ratio of the total number of erroneous keystrokes (*EKS*) to the length of the presented text:

$$EKS\ ER = \frac{EKS}{|P|} \times 100\%. \quad (5)$$

EKS can be derived using the equation: $EKS = INF + IF$.

5) Total Error Rate (Total ER)

Total Error Rate (*Total ER*) is a unified method that combines the effect of accuracy during and after text entry [2]. This metric measures the ratio of the total number of incorrect and corrected characters, which is equivalent to the total number of erroneous keystrokes, to the total number of correct, incorrect, and corrected characters:

$$Total\ ER = \frac{INF+IF}{C+INF+IF} \times 100\%. \quad (6)$$

C. Issues with Error Rate Metrics

The two most widely used error metrics, *ER* and *MSD ER*, can be considered to be almost equivalent [2]. However both do not consider the cost of error correction but only the errors still present in the transcribed text. This can make these two metrics misleading. For example, if all the erroneous character were corrected in the transcribed text, these two metrics will report the same as if the text was entered error free from the start. In other words, they do not consider the effort that was put into correcting errors. *KSPC*, on the other hand, considers the cost of committing errors and fixing them, but does not provide any way of separating these two quantities. Nevertheless, there is an (approximately) inverse relationship between *KSPC* and *ER* respectively *MSD ER*. However, there is no obvious way of combining these measures into an overall error rate [2]. *EKS ER* also considers the cost of committing errors, but fails to show an accurate picture when the transcribed text contains erroneous characters. This is because this metric considers the length of the presented text instead of the total effort to enter the text. Therefore, *EKS ER* is usually used when the final transcribed text was kept error free by forcing the participants to correct each error. *Total ER* overcomes this shortcoming by computing the ratio between the total number of incorrect and corrected characters and the total effort to enter the text, providing more insight into the behaviors of the participants. This makes *Total ER* the most powerful error rate metric at the present time.

III. Mobile Text Entry Method Performance

We collected data for seven important text entry methods: two full-length QWERTY keyboards (standard, and projection), two reduced-size QWERTY keyboards (thumb, and soft); two keypads (Twiddler, and standard 12-key keypad), and one stylus based text entry method (Graffiti).

A. Data Collection

We took a few precautions while collecting data to ensure that our results are solid. We ignored all papers that do not provide complete data about the experiment, use unorthodox performance metrics, or do not follow standard empirical experiment procedures. If a paper used unusual metrics, but provided enough data to permit a conversion into standard metrics, we included the paper. We also did not include pilot studies, non-English or numerical character-based studies, and studies that were carried out with less than five participants per method. This eliminated a substantial number of publications from consideration, but one cannot perform cross-study comparison without some guarantee on the validity of the results and without comparison points.

Most of the surveyed experiments were conducted using MacKenzie and Soukoreff's short English phrases [5] as presented text. A few studies on mobile keypads were conducted using SMS-style phrases [6], which is the kind of text usually entered on hand-held devices. Some experiments use both phrase sets [7]. One study was conducted using phrases *with* and *without* numerical and special characters [8], and we considered only the later data points in our survey.

All the articles we surveyed used the *WPM* metric to measure typing speed. The use of the –1 in the numerator of (1) was specifically mentioned in some articles [9, 10], others, however, did not mention it.

B. Recalculating the Metrics

Although most of our surveyed articles included the data we were looking for, in a few cases we had to derive them from other data. While experimenting on mobile keypads, James and Reischel [7] did not measure errors in any standard metric. They provided the total number of errors in each dataset, but again did not elaborate on how they counted errors. For example, assuming that "abc" was discovered as "acb" in the transcribed text, it is not clear if that was counted as a single or multiple errors. However, we recalculated *EKS ER* from their paper by using (5), using the total number of errors as *EKS*. The resulting error rate was unusually high for the multi-tap technique: 16.6% resp. 29.7% for *novice* and *expert* users. This underlines the importance of using a well-defined error counting methodology. A few articles [11, 12] did not provide the average or individual session error rates in numerical form but in graphs. In those cases we manually measured the data from the graphs to derive numerical error rates. McDermott-Wells [13] did not provide average error rates, but presented exhaustive data on different sessions. From the session data recalculating average error rates was easy. MacKenzie and Read [14] performed a series of mock-up studies, where no real device was used, to determine only text entry speed. This assumes that no errors were made, as it was not possible to track errors. We still considered their data for our survey.

C. Error Correction Conditions

We observed in our survey that text entry experiments are conducted with one of three error correction conditions:

1) None: In this condition, participants are not allowed to correct errors. As a result, the final transcribed text contains only uncorrected errors. Usually *ER* or *MSD ER* metrics are used to measure error rates.

2) Recommended: In this condition, participants are recommended to correct errors as they identify them. Thus, the final transcribed text contains both corrected and uncorrected errors. *Total ER* is usually used to measure error rates.

3) Forced: Here, participants are forced to correct each error to keep the transcribed text error free. Therefore, the final transcribed contains only corrected errors. *Total ER* is usually used to measure error rates, although some researchers keep a separate count of erroneous keystroke to measure *EKS ER*.

D. Survey Results

Table I presents the complete result of our survey.

TABLE I. Text Entry Method Performance From Literature

Text Entry Methods	Ref	Participants		Text Entry Metrics			
		Expertise	#	*Error Correction Condition*	*Error Metric*	*Error Rate*	*WPM*
QWERTY	[15]	Average	11	*None*	*ER*	1.80	64.80
	[16]	Average	14	*Recommended*	×	×	86.87
Projection	[15]	Average	11	*None*	*ER*	3.70	46.60
Thumb Mini-QWERTY	[15]	Average	11	*None*	*ER*	2.20	27.60
	[16]	Expert	7	*Recommended*	*Total ER*	8.32	61.44
	[16]	Expert	7	*Recommended*	*Total ER*	8.32	58.61
	[17]	Expert	8	*Recommended*	*Total ER*	6.70	55.77
Stylus-Based Graffiti	[15]	Average	11	*None*	*ER*	13.60	14.00
	[8]	Average	12	*Recommended*	*Total ER*	19.35	9.24
Soft/Virtual Stylus	[8]	Average	12	*Recommended*	*Total ER*	4.11	13.64
	[13]	Average	7	*Recommended*	*Total ER*	7.40	21.65
	[14]	Average	12	×	×	×	34.50
	[14]	Average	24	×	×	×	28.10
	[14]	Average	12	×	×	×	26.50
Twiddler	[12]	Novice	10	*Recommended*	*Total ER*	4.35	26.20
	[18]	Expert	5	*Recommended*	*Total ER*	6.20	37.30
Mobile 12-Key Multi-tap	[9]	Novice	5	*Forced*	*EKS ER*	2.60	10.11
	[9]	Novice	5	*Forced*	*EKS ER*	4.60	10.33
	[10]	Average	10	*Forced*	*EKS ER*	3.00	10.11
	[6]	Novice	5	*Forced*	*EKS ER*	2.53	7.61
	[7]	Novice	10	*Recommended*	*EKS ER*	16.60	7.98
	[7]	Expert	10	*Recommended*	*EKS ER*	29.70	7.93
	[11]	Average	10	*Forced*	*EKS ER*	4.70	15.50

means the total number of participants, and × means data were not provided in the literature.

1) Average Entry Speed

The standard QWERTY keyboard is the fastest of all methods with an average of 75.85 *WPM* (*SD* = 15.61), while the multi-tap phone keypad is the slowest with an average of 9.94 *WPM* (*SD* = 2.72). Amongst QWERTY type keyboards, the thumb keyboard is the second fastest alternative with an average of 50.86 *WPM* (*SD* = 15.68), surprisingly faster than the full-size projection keyboard. It is worth mentioning at this point that both theoretical and empirical [19] evidence shows that the size of keyboard layout does not have a noticeable

impact on performance. Therefore using different sized keyboards to compare different text entry methods makes sense. Twiddler tops both soft QWERTY and stylus-based Graffiti keyboards with an average of 31.75 *WPM* (*SD* = 7.85), becoming the fastest non-QWERTY text entry method. Fig. 1 presents the average entry speed for our surveyed text entry methods. There was not enough data to calculate a standard deviation (*SD*) for the projection keyboard.

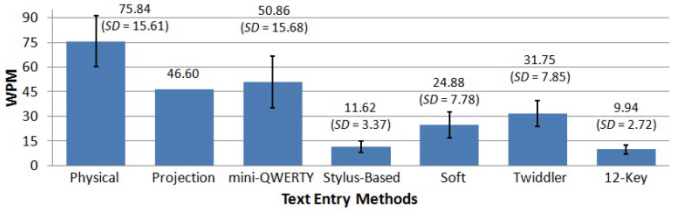

Figure 1. Average *WPM* for text entry methods.

2) *Average Error Rate*

The standard QWERTY keyboard has the lowest error rate with an average of 1.8%, while the stylus-based keyboard has the highest with an average of 19.72% (*SD* = 6.31). Fig. 2 presents the average error rates for our surveyed methods. Unfortunately there wasn't sufficient data to calculate standard deviation (*SD*) for the QWERTY and projection keyboards.

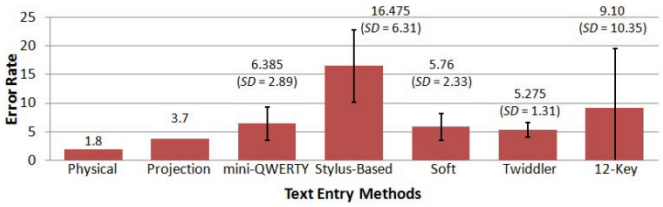

Figure 2. Average error rate for text entry methods.

3) *Data Analysis*

Looking at Fig. 1 and 2 one can observe an approximately indirect relationship between error rate and text entry speed. High error rates imply low *WPM* and vice versa. As most of our surveyed experiments used participants with novice or average expertise, we can conjecture that the error rate affects the speed of text entry (for novice or average users). However, we do not claim that we can explain this relationship fully, as the relationship is far from direct.

Above we discussed that text entry experiments are usually conducted with one of three error correction conditions: *none*, *recommended*, and *forced*. This poses the interesting question if these different conditions have any effect on text entry metrics. We address this question in our next section.

IV. An Experiment

The main purpose of this experiment was to observe if different error correction conditions have an effect on the various text entry measures. We also wanted to investigate the relationship between different error metrics.

A. *Apparatus*

We used a Compaq *KB-0133* QWERTY keyboard and a 19" CRT monitor at 1280×960 for our study. A Java program logged all key presses with timestamps during text entry and calculated user performance directly. We used 15 point Tahoma font on the screen to present text.

B. *Participants*

12 participants from the university community, aged from 22 to 45 (average 27 years), took part in the experiment. All of them were touch typists and proficient in the English language (native speakers, or had spent at least 5 years in an English speaking environment). 9 of our participants were male and 3 female; all of them were right-hand mouse users. They all received a small compensation for their participation.

C. *Procedure*

During the experiment, participants entered short English phrases from MacKenzie and Soukoreff's set [5]. We chose this corpus because of its high correlation with the letter frequency in the English language. Moreover, these phrases are widely used in recent text entry studies, which makes our work comparable with others'. Participants were selected to be touch typists and fluent English speakers to minimize the effect of learning during the experiment. Towards this, anybody who could not achieve an average typing speed of 50 *WPM* on three phrases on a full QWERTY keyboard was excluded from the experiment. Phrases were shown to the participants on the screen in a dialog. They were asked to take the time to read and understand the phrases, to enter them as *fast* and *accurate* as possible, and to press the *enter* key when they were done to see the next phrase. Timing started from the entry of the first character and ended with the last (the character before the *enter* key press). We also informed them that they could rest either between blocks, sessions, or before typing a phrase.

During the *none* condition, participants were asked not to correct any error. They were instructed to ignore errors and carry on if they noticed errors in their typing. For this condition, we disabled all edit functions, modifier, and navigation keys, and mouse operations that could correct errors. During the *recommended* condition, participants were asked to work normally. That is, they correct their errors as they notice them. They were also informed that they could use any edit functions, modifier, navigation keys, or the mouse to correct their errors. During the *forced* condition, we used an error notification function to inform participants of their errors. When an erroneous character was entered the application made a "beep" noise and the input text field turned red. Participants were instructed to take the necessary action(s) to correct that erroneous character before proceeding.

D. *Design*

We used a within-subjects design for the three error correction conditions. There were 3 sessions. In each session participants were asked to complete 9 blocks (3 blocks per condition) containing 20 phrases (excluding practice phrases). Participants were randomly assigned into 3 groups in a 3×3 Latin Square to avoid asymmetric skill transfer.

E. *Results*

The 12 participants took an average of 6.32 minutes for each session, 18.96 minutes for all three sessions, and about 30 minutes for the whole experiment including the demonstration,

and breaks. The highest and lowest average type speeds for our participants were 121 and 55 *WPM*.

1) WPM Analysis

Our hypothesis was that the *none* error correction condition would show higher *WPM* rates than the *recommended* and *forced* ones. It only seemed natural before the experiment that entering error free phrases would require more time, as the measure of time for the former condition would be the sum of the text entry time and the error correction time. Surprisingly, our data did not support this hypothesis. An ANOVA analysis established that there is no significant effect of different error correction conditions in text entry experiments on *WPM* ($F_{2,11} = 3.11$, ns). Average *WPM* for the *none*, *recommended*, and *forced* conditions were 81.82 ($SD = 21.71$), 80.65 ($SD = 19.96$), and 78.56 ($SD = 17.95$), respectively.

2) KSPC Analysis

An ANOVA analysis showed that there was a significant effect of error correction conditions on *KSPC* ($F_{2,11} = 28.46$, $p < .0001$). A Tukey-Kramer multiple-comparison test showed that the *recommended* and *forced* conditions had significantly higher *KSPC* than the *none* condition. On average, these two conditions had 8.15% ($SD = 4.41$) and 9.02% ($SD = 4.63$) more *KSPC* than the *none* condition, see Fig. 3.

Figure 3. Average *KSPC* for error correction conditions.

3) EKS ER and Total ER Analysis

An ANOVA analysis showed that there was a significant effect of the different error correction conditions on both *EKS ER* ($F_{2,11} = 8.42$, $p < .005$) and *Total ER* ($F_{2,11} = 9.77$, $p < .001$). A Tukey-Kramer multiple-comparison test showed that the *recommended* and *forced* conditions had significantly higher *EKS ER* and *Total ER* than the *none* condition. On average these two conditions had 66.49% ($SD = 47.33$) respectively 62.38% ($SD = 42.38$) more *EKS ER*, and 50.54% ($SD = 36.83$) respectively 43.10% ($SD = 26.37$) more *Total ER* than the *none* condition. Fig. 4 shows the average *EKS ER* and *Total ER* for each condition.

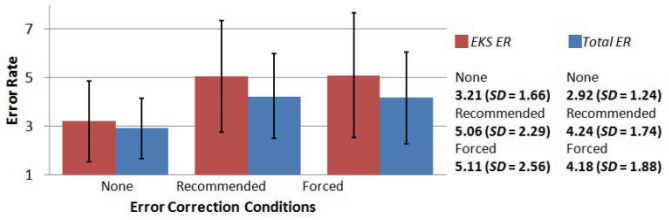

Figure 4. Average *EKS ER* and *Total ER* for error correction conditions.

4) ER and the MSD ER Analysis

As we made sure the final transcribed text was error free by forcing our participants to correct each error during the *forced* condition. *ER* and the *MSD ER* measured zero errors for this condition. Hence, we compared these two metrics only for the *none* and *recommended* conditions. An ANOVA analysis showed that there was a significant effect of error correction conditions on both *ER* ($F_{1,11} = 38.91$, $p < .0001$) and *MSD ER* ($F_{1,11} = 38.65$, $p < .0001$). Fig. 5 illustrates the average *ER* and the *MSD ER* for these conditions, where we can see that these measures are very close. We observed that the *recommended* condition had 18.40% ($SD = 20.80$), and 18.41% ($SD = 20.95$) lower *ER* and *MSD ER* than the *none* condition.

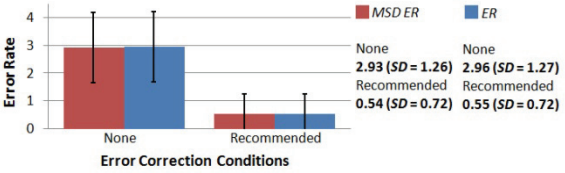

Figure 5. Average *ER* and the *MSD ER* for error correction conditions.

5) Visual Scan Time

Our participants had to press the *enter* key after they were done typing to see the next phrase. We observed that participants usually took time to quickly scan through the typed phrase before they pressed the *enter* key. On average they took 298 ms ($SD = 298$) before pressing the *enter* key: 294 ms ($SD = 131$) during the *none*, 348 ms ($SD = 493$) during the *recommended*, and 252 ms ($SD = 82$) during the *forced* condition. An ANOVA analysis showed that there was no significant effect of error correction conditions on the visual scan time ($F_{2,11} = 0.39$, ns). We were also unable to find any obvious relationship between the visual scan time, the length of the transcribed text, or the typing speed.

V. Discussion

A. Entry Speed

In our experiment it became clear that error correction conditions do not have any significant effect on *WPM* for expert users on a QWERTY keyboard. We see two potential reasons for this result. First, the *WPM* calculation considers all the characters in the transcribed text, not only the correct ones. This means, incorrectly inputted characters during the *none* and *recommended* conditions were also counted for the *WPM* calculation. Second, we noticed that during the *none* condition sometimes typists instinctively tried to correct their errors before they remembered that they could not. Such a failed error correction attempt takes a bit of time, as participants need to mentally recover and resume the original task. Again, during the *recommended* condition participants tended to correct their errors almost the moment they made them (i.e. character level error correction), making this condition close to the forced condition. An ANOVA analysis confirmed that there was no significant difference between the number of edit keystrokes in the *recommended* and *forced* ($F_{1,11} = 0.65$, ns), and the edit keystrokes did not significantly differ across sessions. We did not find any relationship between the typists' entry speed and their instinctive attempt to correct errors. Note that our participants were all expert typists, hence novices may show different behavior. Fig. 6 shows the average edit keystrokes for each condition.

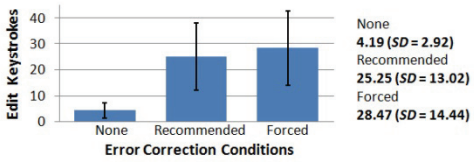

Figure 6. Average edit keystrokes for error correction conditions.

B. Corrective Keystrokes

We noticed that typists almost exclusively used the *backspace* key while entering text, although we informed them beforehand that they could use the keyboard shortcuts or the mouse (a mouse click was considered as a single keystroke) to perform edit operation if they wanted to. In our experiment 99% of the all edit keystrokes were *backspace*.

C. Error Rate Metrics

Our result showed that there was a significant effect of error correction conditions on all major error metrics. This finding underlines the importance of presenting error rate measures along with the *WPM* measure when comparing new text entry techniques. Our result also showed that the *recommended* and *forced* conditions had significantly higher *KSPC* than the *none* condition. The reason behind this behavior is that the *KSPC* measure compares the input stream and the transcribed text, not the presented text. As both the input stream and transcribed text contains erroneous characters, the *KSPC* value always remains lower for the QWERTY keyboard. Yet, the *KSPC* value is never one because of the presence of keystrokes belonging to the edit, modifier, and navigation keys in the input stream. Our result indicated that the *ER* and *MSD ER* measures are almost equivalent. Other error rate measures, however, do not seem to have any simple relationship that would enable conversion from one to another.

D. Visual Scan Time and the Phrase Set

We found that there is no significant effect of different error correction conditions on the visual scan time and there is no obvious relationship between the visual scan time and the length of the transcribed text or the typing speed. But note that in our experiment the average length of the presented text was 28.88 ($SD = 1.23$), and all of our participants were expert typists. This may differ in other scenarios.

A small issue that occurred during the experiment was that the phrase set [5] we used for our experiment used American English spelling (e.g., flavored instead of flavoured, etc.), whereas all of our participants were familiar with the British English spelling. During the *forced* error correction condition this occasionally caused irritation for our participants, as they could not easily find the error at a glance.

VI. Future Work

In future we plan to investigate if it is possible to design a mathematical model to predict the cost of error correction for various text input devices. We would also like to examine if different user expertise and phrase length have any effect on instinctive error correction and visual scan time.

VII. Conclusion

A survey of seven well-known text entry methods was presented. We reported and analyzed our surveyed data to make it easier to compare these methods. We also presented the results of a study to determine the effect of error correction conditions on the most common text entry performance metrics.

References

[1] J. O. Wobbrock, "Measures of text entry performance," in Text Entry Systems: Mobility, Accessibility, Universality, 1st ed., I. S. MacKenzie and K. Tanaka-Ishii, Eds., San Francisco, CA: Morgan Kaufmann, 2007, pp. 47-74.

[2] R. W. Soukoreff and I. S. MacKenzie, "Metrics for text entry research: an evaluation of MSD and KSPC, and a new unified error metric," in Proc. CHI, 2003, pp. 113-120.

[3] H. Yamada, "A historical study of typewriters and typing methods: From the position of planning Japanese parallels," The Journal of Information Processing, vol. 2, pp. 175-202, 1980.

[4] R. W. Soukoreff and I. S. MacKenzie, "Measuring errors in text entry tasks: an application of the Levenshtein string distance statistic," in Proc. CHI Extended Abstracts, 2001, pp. 319-320.

[5] I. S. MacKenzie and R. W. Soukoreff, "Phrase sets for evaluating text entry techniques," in Proc. CHI Extended Abstracts, 2003, pp. 754-755.

[6] H. Ryu and K. Cruz, "LetterEase: improving text entry on a handheld device via letter reassignment," in Proc. OZCHI, 2005, pp. 1-10.

[7] C. L. James and K. M. Reischel, "Text input for mobile devices: comparing model prediction to actual performance," in Proc. CHI, 2001, pp. 365-371.

[8] T. Költringer and T. Grechenig, "Comparing the immediate usability of graffiti 2 and virtual keyboard," in Proc. CHI Extended Abstracts, 2004, pp. 1175-1178.

[9] D. Wigdor and R. Balakrishnan, "A comparison of consecutive and concurrent input text entry techniques for mobile phones," in Proc. CHI, 2004, pp. 81-88.

[10] D. Wigdor, and R. Balakrishnan, "TiltText: using tilt for text input to mobile phones," in Proc. UIST, 2003, pp. 81-90.

[11] I. S. MacKenzie, H. Kober, D. Smith, T. Jones, and E. Skepner,. 2001. "LetterWise: prefix-based disambiguation for mobile text input", in Proc. UIST, 2001, pp. 111-120.

[12] K. Lyons, T. Starner, D. Plaisted, J. Fusia, A. Lyons, A. Drew and E. Looney, "Twiddler typing: one-handed chording text entry for mobile phones," In Proc. CHI, 2004, pp. 671-678.

[13] P. McDermott-Wells, "Evaluation of three stylus-based text entry methods on a pocket PC™ mobile device," in Proc. SoutheastCon, 2006, pp. 228-234.

[14] I. S. MacKenzie and J. C. Read, "Using paper mockups for evaluating soft keyboard layouts," in Proc. CASCON, 2007, pp. 98-108.

[15] H. Roeber, J. Bacus and C. Tomasi, "Typing in thin air: the canesta projection keyboard - a new method of interaction with electronic devices," in Proc. CHI Extended Abstracts, 2003, pp. 712-713.

[16] E. Clarkson, J. Clawson, K. Lyons and T. Starner, "An empirical study of typing rates on mini-QWERTY keyboards," in Proc. CHI Extended Abstracts, 2005, pp. 1288-1291.

[17] J. Clawson, K. Lyons, T. Starner and E. Clarkson, "The impacts of limited visual feedback on mobile text entry for the Twiddler and mini-QWERTY keyboards," in Proc. ISWC, 2005, pp.170-177.

[18] K. Lyons, D. Plaisted and T. Starner, "Expert chording text entry on the Twiddler one-handed keyboard," In Proc. ISWC, 2004, pp. 94-101.

[19] I. S. MacKenzie and S. X. Zhang, "An empirical investigation of the novice experience with soft keyboards," Behaviour & Information Technology, vol. 20, pp. 411-418.