INVESTIGATION OF KEYBOARD AND SPEECH BASED TEXT ENTRY ON MOBILE DEVICES

Shyam Mehraaj Reyal

A Thesis Submitted for the Degree of PhD
at the
University of St Andrews

2019

Investigation of Keyboard and Speech Based Text Entry on Mobile Devices

Shyam Mehraaj Reyal

University of St Andrews

This thesis is submitted in partial fulfilment for the degree of

Doctor of Philosophy (PhD)

at the University of St Andrews

October 2018

Candidate's declaration

I, Shyam Mehraaj Reyal, do hereby certify that this thesis, submitted for the degree of PhD, which is approximately 51,000 words in length, has been written by me, and that it is the record of work carried out by me, or principally by myself in collaboration with others as acknowledged, and that it has not been submitted in any previous application for any degree.

I was admitted as a research student at the University of St Andrews in September 2012.

I received funding from an organisation or institution and have acknowledged the funder(s) in the full text of my thesis.

01-10-2018
Date   Signature of candidate

Supervisor's declaration

I hereby certify that the candidate has fulfilled the conditions of the Resolution and Regulations appropriate for the degree of PhD in the University of St Andrews and that the candidate is qualified to submit this thesis in application for that degree.

01-10-2018
Date   Signature of supervisor

Permission for publication

In submitting this thesis to the University of St Andrews we understand that we are giving permission for it to be made available for use in accordance with the regulations of the University Library for the time being in force, subject to any copyright vested in the work not being affected thereby. We also understand, unless exempt by an award of an embargo as requested below, that the title and the abstract will be published, and that a copy of the work may be made and supplied to any bona fide library or research worker, that this thesis will be electronically accessible for personal or research use and that the library has the right to migrate this thesis into new electronic forms as required to ensure continued access to the thesis.

I, Shyam Mehraaj Reyal, confirm that my thesis does not contain any third-party material that requires copyright clearance.

The following is an agreed request by candidate and supervisor regarding the publication of this thesis:

Printed copy

No embargo on print copy.

Electronic copy

Embargo on all of electronic copy for a period of 3 years on the following ground(s):

- Publication would preclude future publication

Supporting statement for electronic embargo request

The work in some of the thesis chapters e.g. chapter 5 and 6, are part of an ongoing manuscript for publication to IUI 2019, CHI 2020, or a future conference. Publishing the electronic copy of the thesis would hinder these as the research ideas could be usurped and use elsewhere before the paper is published, and if submitted would conflict with the blind-peer-review processes incorporated in those conferences.

Title and Abstract

- I agree to the title and abstract being published.

01-10-2018 Date	Signature of candidate
01-10-2018 Date	Signature of supervisor

Underpinning Research Data or Digital Outputs

Candidate's declaration

I, Shyam Mehraaj Reyal, understand that by declaring that I have original research data or digital outputs, I should make every effort in meeting the University's and research funders' requirements on the deposit and sharing of research data or research digital outputs.

01-10-2018
Date Signature of candidate

Permission for publication of underpinning research data or digital outputs

We understand that for any original research data or digital outputs which are deposited, we are giving permission for them to be made available for use in accordance with the requirements of the University and research funders, for the time being in force.

We also understand that the title and the description will be published, and that the underpinning research data or digital outputs will be electronically accessible for use in accordance with the license specified at the point of deposit, unless exempt by award of an embargo as requested below.

The following is an agreed request by candidate and supervisor regarding the publication of underpinning research data or digital outputs:

Embargo on all of electronic files for a period of 3 years on the following ground(s):

- Publication would preclude future publication

Supporting statement for embargo request

For the same reasons for requesting embargo of electronic thesis

Title and Description

- I require an embargo on the title and description

01-10-2018
Date Signature of candidate

01-10-2018
Date Signature of supervisor

Acknowledgements

I wish to convey my thanks to everyone who advised and supported me throughout this work presented in this thesis.

My advisors & collaborators:

- Dr Per Ola Kristensson (primary advisor)
  University Reader, University of Cambridge, UK

- Dr Mark-Jan Nederhof (secondary advisor)
  Lecturer, University of St Andrews, UK

- Dr Keith Vertanen (external advisor)
  Asst. Professor, Michigan Tech, USA

- Dr Shumin Zhai (collaborator)
  Senior Staff Research Scientist, Google Inc., USA

My Family and Friends:

- My parents
- My wife – Dr Nouchali Bandaranayaka
- My friends

Funding:

- This work was supported by the Scottish Informatics and Computer Science Alliance (SICSA).
- This work was supported by the University of St Andrews (School of Computer Science).
- This work was supported by the University of St Andrews (International Student Hardship Fund).

Abstract

This work presents four in-depth empirical investigations on the performance and user experience of three popular mainstream mobile text entry methods: Touch Typing on a Software Keyboard (STK), the Gesture Typing on a Software Keyboard (SGK), and Speech Based Text Entry. The first and third studies are lab-based longitudinal text entry experiments. In the second and fourth studies, we use a new text entry evaluation methodology based on the experience sampling method (ESM). In the ESM based studies, participants installed an Android app on their own mobile phones that periodically sampled their text entry performance and user experience amid their everyday activities for four weeks. The studies show that text can be entered at an average speed of 24 to 41 WPM using software keyboards, and 49 to 59 WPM using speech, depending on the method and the user's experience, with 0.9% to 3.6% character error rates remaining for software keyboard and 3.0% to 5.8% for speech. Error rates of SGK and speech based input are a major challenge; and reducing out-of-vocabulary errors is particularly important. Both typing and speech have strengths, weaknesses, and different individual awareness and preferences. Two-thumb touch typing in a focused setting is particularly effective on STK, whereas one-handed SGK typing with the thumb is particularly effective in more mobile situations. Speech is more effective when convenience and constraints take priority, whereas typing is more preferable in public – due to social concerns, network latency issues and background noise. When exposed, users showed a trend to migrate from STK to SGK. We also conclude that studies in the lab and in the wild can both be informative to reveal different aspects of keyboard and speech based text entry, but used in conjunction is more reliable in comprehensively assessing input technologies of current and future generations.

University of St Andrews | FOUNDED 1413

University Teaching and Research Ethics Committee

30 January 2019

Dear Shyam,

Thank you for submitting your ethical application, which was considered by the School of Computer Science Ethics Committee on Wednesday 6th June, where the following documents were reviewed:

1. Ethical Application Form
2. Participant Information Sheet
3. Anonymous Consent Form
4. Debriefing Form

The School of Computer Science Ethics Committee has been delegated to act on behalf of the University Teaching and Research Ethics Committee (UTREC) and has granted this application ethical approval. The particulars relating to the approved project are as follows -

Approval Codes:	CS13888 CS10246 CS10251	**Approved on:**	23.08.18	**Approval Expiry:**	23.08.2023
Project Title:	Investigation of Typing vs Speech on Mobiles In The Lab and In The Wild				
Researcher(s):	Shyam Reyal				
Supervisor(s):	Mark-Jan Nederhof				

Approval is awarded for five years. Projects which have not commenced within two years of approval must be re-submitted for review by your School Ethics Committee. If you are unable to complete your research within the five year approval period, you are required to write to your School Ethics Committee Convener to request a discretionary extension of no greater than 6 months or to re-apply if directed to do so, and you should inform your School Ethics Committee when your project reaches completion.

If you make any changes to the project outlined in your approved ethical application form, you should inform your supervisor and seek advice on the ethical implications of those changes from the School Ethics Convener who may advise you to complete and submit an ethical amendment form for review.

Any adverse incident which occurs during the course of conducting your research must be reported immediately to the School Ethics Committee who will advise you on the appropriate action to be taken.

Approval is given on the understanding that you conduct your research as outlined in your application and in compliance with UTREC Guidelines and Policies (http://www.st-andrews.ac.uk/utrec/guidelinespolicies/). You are also advised to ensure that you procure and handle your research data within the provisions of the Data Provision Act 1998 and in accordance with any conditions of funding incumbent upon you.

Yours sincerely

Wendy Boyter

School Ethics Committee Administrator

1

Introduction

1.1 Motivation

It is fair to say that mobile devices have indeed taken over our lives. We access everyone and everything using our mobile device in our day to day life – from our phone contacts, our to-do lists, our calendars, and our email to our bank accounts. We use our mobile devices to browse the internet, keep in touch with colleagues, friends and loved ones, use it to click pictures, share them on social media, and keep up to date with others' social lives. We now even use it as a replacement for payment mechanism such as credit or debit cards, and use it as mechanisms to allow second-level authentication into our secure accounts such as banks, PayPal, and other financial portals. It is fair to claim there is no other device as close to the human being as their mobile phone. So what is the most significant or important feature of a mobile device? Two decades ago, one would say "calling" is the most important feature - but it is not the case anymore. With the widespread behaviour of "texting" and the use of "messaging apps", it is obvious that entering text is the most important commodity feature of a mobile device. According a study from (Nielson, 2005), a teenager on average sends over 3,000 messages per month, or more than six texts per waking hour.

In the recent years, there has been a significant improvement of entering text on a mobile device (covered in Chapter 2 – Literature Review), and this thesis aims to further shed light on this matter by the evaluating the performance and user experience of current "state of the art" text entry mechanisms on mobile devices.

1.2 Central Idea and Research Questions

The central theme surrounding this thesis is that we can understand the factors enhancing or limiting the performance and user experience of entering text on mobile devices by evaluating them in both a lab setting and “in the wild” (Kjeldskov & Skov, 2014) bringing us to this central idea:

> **"Controlled experimental A/B lab comparisons and ESM-based in-the-wild studies both inform similar and complementary aspects of the text entry experience. However, when used in conjunction, they are capable of more comprehensively assessing the complete text entry user experience.”**

In this central idea, we use the ISO standard definition of user experience "a person's perceptions and responses that result from the use or anticipated use of a product, system or service" (“ISO 9241-210:2010 - Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems,” 2010).

This thesis attempts to further shed light on this central idea by answering the 4 main research questions.

1. What are the factors that affect the performance and user experience of mobile text entry using a Smart Touch Keyboard (STK) and a Smart Gesture Keyboard (SGK) in a lab setting?

2. What are the factors that affect the performance and user experience of mobile text entry using a Smart Touch Keyboard (STK) and a Smart Gesture Keyboard (SGK) outside a lab setting a.k.a. in the wild?

3. What are the factors that affect the performance and user experience of mobile text entry using a keyboard vs speech input in a lab setting?

4. What are the factors that affect the performance and user experience of mobile text entry using a keyboard vs speech input outside a lab setting a.k.a. in the wild?

1.3 Chapter Outline

The remainder of this thesis is structured as follows. This thesis is a monograph and should be read in the given order of Chapters 1 to 8.

- **Chapter 1 – Introduction**
  The rest of this chapter is dedicated to describing some of the common terms used throughout this thesis

- **Chapter 2 – Literature Review**
  This chapter contains a thorough review of the work in the fields of mobile text entry, speech recognition, and evaluation methods for assessing the performance and user experience of same

- **Chapters 3 – 6**
  Each of these chapters aims to answer the research questions 1-4 mentioned above. Each chapter outlines an experiment – the motivations, hypotheses, methodology, apparatus, participants, procedure, results and analyses.

- **Chapter 7 – Discussion**
  This chapter contains a thorough discussion of the results and analyses from chapters 3-6.

- **Chapter 8 - Conclusion**
  This chapter makes concluding statements and remarks about the findings made in this thesis.

1.4 Text Entry on Mobile Devices

Text entry on touchscreen mobile devices is typically carried out using one of two text entry methods. They are the Smart Touch Keyboard (STK) and the Smart Gesture Keyboard (SGK).

1.4.1 The Smart Touch Keyboard (STK)

The first involves typing using a single finger or both thumbs on a touch tapping QWERTY keyboard. In this thesis we will call this method Smart Touch Keyboard (STK). Modern STKs perform automatic typing correction and allow users to choose among word predictions – see Figure.

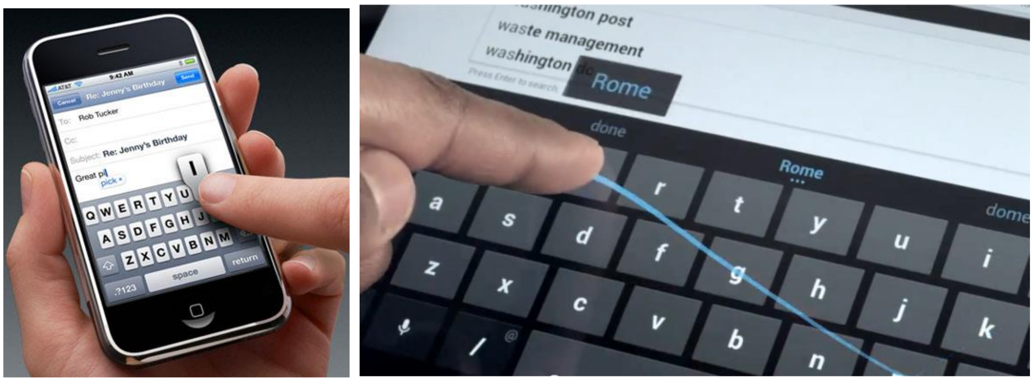


Figure 1 - Smart Touch Keyboard vs Smart Gesture Keyboard

https://www.tested.com/tech/smartphones/197-whats-the-best-way-to-type-on-a-smart-phone/

https://www.xda-developers.com/android-4-2-keyboard-with-gesture-typing-leaked/

1.4.2 Smart Gesture Keyboard (SGK)

An alternative text entry method is the Smart Gesture Keyboard (SGK) (Shumin Zhai & Kristensson, 2012). To write on a SGK a user slides a finger across the touchscreen keyboard. For example, to write the word “the” the user may land on the T key, slide to the H key, continue to the E key, and then lift up the finger – see Figure 1 (right). This produces a gesture that is recognized by the system and is pattern matched to find the word whose trace on the keyboard most resembles the user entered gesture. This relatively novel “word gesture” keyboard paradigm has appeared in commercial products commercial such as ShapeWriter (P.-O. Kristensson, 2007), Swype, T9 Trace (Endgadget, n.d.), and Google Keyboard (Google, 2018c) on Android.

The algorithms used in today's state of the art commercially deployed keyboards have not been published to our knowledge, although the Android STK's source code is available from the Android Open Source Project (AOSP) (Google, 2018b). Note that this is different from Google’s proprietary release of their own keyboard a.k.a. the GBoard (Google, 2018c) According the AOSP code and the published literature based on research prototypes and experiments (see Chapter 2 – Literature Review), STKs and SGKs in principle decode user' intended words by combining lexical or language model information with user's imprecise input. The size of these keyboards’ vocabulary is important to error rate. The Android AOSP keyboard has a lexicon size of 160K words for English.

1.4.3 Keyboard Layout

For all the studies in this thesis, we used the QWERTY keyboard layout for both STK and SGK, as it is considered the most widely used (Gizmodo, 2014). Other keyboard variants (such as DVORAK and ATOMIK) are explored in the Literature Review section, however due to its widespread popularity and ease to recruit participants we used QWERTY.

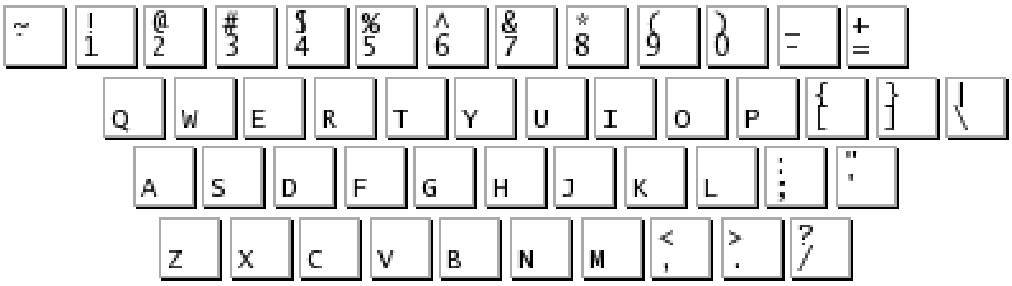

~ ! 1 @ 2 # 3 $ 4 % 5 ^ 6 & 7 * 8 (9) 0 - + =
Q W E R T Y U I O P { [}] | \
A S D F G H J K L : ; " '
Z X C V B N M < , > . ? /

Figure 2 - QWERTY Keyboard Layout - https://en.wikipedia.org/wiki/Keyboard_layout

The above figure shows an “unbiased” QWERTY keyboard layout (unbiased meaning not implementation specific for PC or Mac) showing the key placements for each letter of the keyboard, numbers from 1-9 and 0, and some most commonly used symbols when typing using English.

1.4.3.1 QWERTY on mobile

The QWERTY layout in the above image is not implemented as is on the mobile device due to lack of space on the limited size mobile screen. Therefore, a user has to ‘switch’ between multiple views of the keyboard to type letters (A-Z, a-z), numbers (1-9), and symbols. This is shown in the next figure.

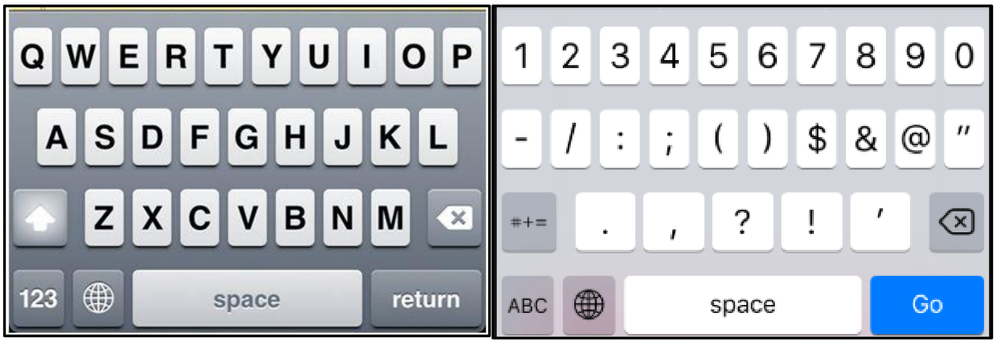


Figure 3 - Character vs Numeric and Symbols View

Therefore, for the purpose of the experiments outlined in this thesis, we ensured the users only characters (A-Z) – which will be explain in detail in Chapters 3-6 under Materials > Phrase Set. This was done in order to prevent switching between multiple views and thereby introducing delays and more confounding variables – as also described in the aforementioned chapters.

1.4.4 Speech Based Text Entry

In addition to keyboard based text entry, in the recent years we identify speech input as a main steam input mechanism in smart devices such as mobiles – such as Google Speech Engine for Android and Siri for iOS, smartwatches, smart-home devices such as Amazon Dot, Google Home, etc., Smart Speakers, and automobile in-car-navigation and entertainment systems such as Apple Car, Android Auto, and manufacturer dependent systems in brands such as Audi, BMW, Bentley, Mercedes, Tesla and Many more (Google, 2018a).

Using voice commands and using speech recognition for text entry has been deemed as a mechanism where visual and tactile feedback is not a requirement, thereby making it safe to use when driving – whilst in many countries it is illegal to text and drive – and other applications where one's visual attention and focus is needed elsewhere.

1.5 Conventions

A number of conventions have been used throughout this thesis – especially in Chapters 3-6. To simplify understanding of the terms used in those chapters, we hereby provide descriptions of the conventions used. Each of the chapters will individually explain if the convention has been changed or modified to fit the experiment in the specific chapter.

1.5.1 The User | The Participant

These terms will be used interchangeably in this thesis. The user or participant refers to the human user who participates in the experiments outlined in Chapter 3 – 6, in which we evaluate the performance and user experience of STK, SGK, and speech based input.

1.5.2 Transcription Task

For the entirety of this thesis, we measure the performance of mobile text entry using a transcription task, as per the standard found in most text entry literature in the field. The transcription task simply means a participant is given a phrase or sentence to copy. Different tasks that can be used to measure participant performance is outlined in Chapter 2 – Literature Review. We define a "stimulus phrase" and a "response phrase" with regards to each transcription task, as follows.

1.5.2.1 Stimulus Phrase (S_1)

This is the phrase shown to the user, which is expected to be copied.

1.5.2.2 Response Phrase (S_2)

This is the resulting phrase from what the user actually types on the given mechanism of mobile text input.

1.5.2.3 Number of characters in the stimulus phrase (N_1)

This is simply of characters in the stimulus phrase (including spaces).

$$N_1 = count\_characters(S_1)$$

1.5.2.4 Number of characters in the response phrase (N_2)

This is simply of characters in the response phrase (including spaces). We are interested more in the number of characters in the response phrase, as this is what the user actually types, from which typing speed is calculated.

$$N_2 = count\_characters(S_2)$$

1.5.2.5 Typing duration (T)

In line with literature, this is simply the difference between when first and the last keystroke falls, in the case of keyboard input, and the time the user speaks in the case of speech input. The typing duration T is usually a derived measurement based on two other variables T1 (the start of text entry) and T2 (the end of text entry) per attempt.

$$T = T_2 - T_1$$

Capturing T1 and T2 are implementation dependent with based on the text entry mechanism in question, which will be explained individually in Chapters 3-6.

1.5.2.6 The Error (E)

The error is the difference between the stimulus phrase (S1) and the response phrase (S2). To measure the difference between two strings, we use the Levenshtein Distance (Levenshtein, 1966) or L.

Levenshtein distance is named after the Russian scientist Vladimir Levenshtein, who devised the algorithm in 1965. This is a measure of the similarity between two strings, which we will refer to as the source string (s) and the target string (t). The distance is the number of deletions, insertions, or substitutions required to transform s into t. For example,

- If s is "test" and t is "test", then LD(s,t) = 0, because no transformations are needed. The strings are already identical.
- If s is "test" and t is "tent", then LD(s,t) = 1, because one substitution (change "s" to "n") is sufficient to transform s into t.

The greater the Levenshtein distance, the more different the strings are.

1.5.2.6.1 Examples

The Levenshtein distance between "kitten" and "sitting" is 3, since the following three edits change one into the other, and there is no way to do it with fewer than three edits:

- kitten → sitten (substitution of "s" for "k")
- sitten → sittin (substitution of "i" for "e")
- sittin → sitting (insertion of "g" at the end).

1.5.2.6.2 The Algorithm

The algorithm for calculating Levenshtein distance can be outlined as follows, using the Java programming language, as all the software implementations featured on this thesis use Java.

```java
public class EditDistanceRecursive {
   static int calculate(String x, String y) {
        if (x.isEmpty()) {
            return y.length();
        }

        if (y.isEmpty()) {
            return x.length();
        }

        int substitution = calculate(x.substring(1), y.substring(1))
         + costOfSubstitution(x.charAt(0), y.charAt(0));
        int insertion = calculate(x, y.substring(1)) + 1;
        int deletion = calculate(x.substring(1), y) + 1;

        return min(substitution, insertion, deletion);
    }

    public static int costOfSubstitution(char a, char b) {
        return a == b ? 0 : 1;
    }

    public static int min(int... numbers) {
        return Arrays.stream(numbers)
          .min().orElse(Integer.MAX_VALUE);
    }
}
```

1.5.2.7 Entry Rate (WPM)

The entry rate is measured in words per minute (WPM). The standard number of characters per word in the English language is 5 (Mayzner & Tresselt, 1965). Therefore Entry rate is calculated as:

$$WPM = 12 \times \frac{N}{T}$$

Where:

N = is measured in number of characters

T = is measured in seconds

1.5.2.8 Error Rate

The error rate is simply the number of errors as a ratio to the length of the phrase. Here the length is the maximum of the stimulus and response phrases. There are two main conventions of calculating error rate; these are known as Character Error Rate (CER) and Word Error Rate (WER).

1.5.2.8.1 Character Error Rate (CER)

The Character Error Rate (CER) is calculated as follows:

$$CER = \frac{L}{max(N_1, N_2)} \times 100$$

Where:

L = Levenshtein distance (as defined above)

N = number of characters in the phrase (as defined above)

1.5.2.8.2 Word Error Rate (WER)

The Word Error Rate (WER) is calculated in the same principle as Character Error Rate (CER) but by treating an individual word as a unit of measurement, as opposed to a character. We could apply the Levenshtein distance algorithm in the same manner for words instead of characters, and calculate as follows:

$$WER = \frac{L_W}{max(W_1, W_2)} \times 100$$

Where:

L_W = Levenshtein distance considering words as single units

W = total number of words in the phrase

Sometimes, it makes sense to consider the WER than the CER – especially in the case of speech recognition – when the inference is done at a word level as opposed to character level.

1.5.3 Out of Vocabulary Words (OOV)

In simple terms, these are words that would never be candidates suggested on a predictive text input method. In terms of a Smart Touch Keyboard or STK, this would not pose a huge problem as if a word doesn't exist in the prevalent lexicon, e.g. in the case of a proper noun, a name of a person or a place, a user can always go back and correct this by entering each character without any error correction or word prediction. In the case of the Smart Gesture Keyboard, or SGK, this poses a larger problem as words that are not included in the prevalent dictionary will never be suggested as the user types. Therefore the user either has to update the dictionary (which is implementation specific) to include the word he or she wants to type, or revert to a different method of input, i.e. STK, to enter this particular word. This problem also prevails in speech recognition systems. We were interested to find out how the use of OOV words actually affected the typing performance on each input mechanism – in this case the STK and the SGK. How OOV words affect each of the studies is explained in their respective chapters 3-6.

1.5.4 Hand Posture

When typing on mobile devices, it can be seen that users have various different hand postures, which definitely affect their performance - speed and error rate (Goel, Jansen, Mandel, Patel, & Wobbrock, 2013). In this study, we identify 3 possible hand postures

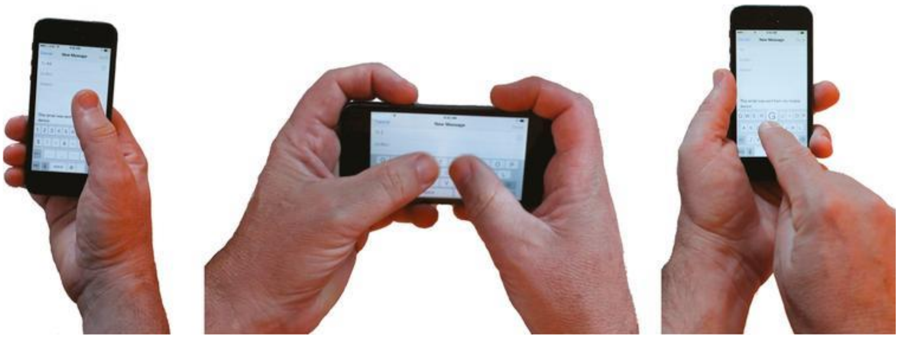


Figure 4 - Hand Postures - https://content.iospress.com/articles/work/wor2159 accessed 30/09/2018

1.5.4.1 Single Finger

This is when the user holds the mobile device in their non-dominant hand, and uses the index finger from the dominant hand to type – touching each key once, in the case of STK, or sliding across the keys, in the case of SGK, with the tip of the index finger. A full visual feedback loop is required when using this hand posture, as the users two hands could operate completely independent of each other.

1.5.4.2 Single Thumb

This hand posture is when a user rests the mobile device on the four fingers and palm of their hand, and using the thumb on the same hand, performs touches (for STK), or gestures (for SGK). An advantage of this approach is that full visual feedback is not required as the mobile phone and the thumb do not operate will full degree of freedom, and the user could reach the mobile keyboard with pivoting from his/her own hand. A disadvantage would be that the mobile device size to hand size ratio would come into play – because a user would need sufficiently large fingers (thumb in this case) to reach the ends of the on screen keyboard. To overcome this difficulty, mobile soft keyboards now provide a one-handed-mode, where the keyboard is made smaller and shifted to one end of the screen so the keys can be reached easily by a user with smaller/shorter fingers.

1.5.4.3 Two Thumbs

In the case of STK, this is very straightforward, as the user holds the mobile in both hands, in portrait or landscape mode, and uses two thumbs to type. This has both the advantages from the previous two hand postures: (a) due to the constrained degree of freedom full visual feedback is not required, and (b) the two thumbs can independently operate on two sides of the keyboard, therefore utilising the entire onscreen keyboard space.

When using SGK, two thumb typing takes on a different form – known as the Bi-Manual Gesture Input (Bi, Chelba, Ouyang, Partridge, & Zhai, 2012) mechanism. The user can perform a gesture to represent parts of the word using either thumb in conjunction with the next, thereby entering the desired word.

1.5.5 Statistical Analyses

All statistical analyses were done using repeated-measures analysis of variance at significance level α=0.05. Bonferroni corrections were used to adjust the significance levels for post-hoc analyses. We report the majority of the statistical results in tables. In the tables "m" is the sample mean, 95% CI means the 95% confidence interval (Z-scores). This is in accordance with standard practice across literature in the field.

1.6 Android or iPhone

All the studies outlined in this thesis are done on the Android platform using Android mobile devices. The main reason for this is SGK's are only supported on Android devices, where as many iOS users do not have any exposure towards it. However, we do recruit iOS users in the studies when we require users with little to no experience on Google Keyboard or SGK. We believe that our results are applicable to a wider population of users and devices as Android mobile devices dominate the world market as shown in the image below. Further, given the open source nature of Android, and supporting a variety of devices, we could evaluate these text input mechanisms on a wide variety of hardware and form factors.

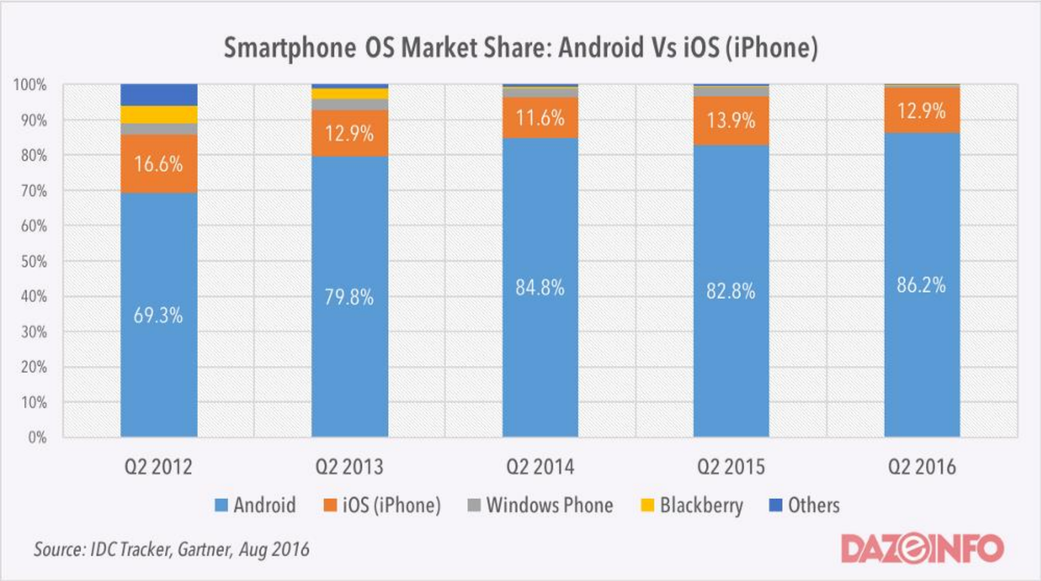


Figure 5 - World Phone Market Distribution

https://android.jlelse.eu/apple-vs-android-a-comparative-study-2017-c5799a0a1683

accessed 30/09/2018

1.7 Perplexity

In this thesis, specifically in chapters 5 and 6, we use the term "perplexity" as a measure of difficulty/complexity of the sentence we require the users to transcribe. In this context, the higher the perplexity, the more complex the sentence is – which we hypothesize that sentences with higher perplexity will yield lower entry rates and/or higher error rates. The formal definition for perplexity has been outlined in the book titled "Speech and Language Processing" (Jurafsky & Martin, 2009), from which we obtain this formulae:

$$PP(W) = P(w_{1,}w_{2,} \ldots w_N)^{-\frac{1}{N}} = \sqrt[N]{\frac{1}{P(w_{1,}w_{2,} \ldots w_N)}}$$

We can use the chain rule to expand the probability of *W*:

$$PP(W) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i|w_1 \ldots w_{i-1})}}$$

1.8 Publications from this thesis

The work and results presented in this thesis has been published or awaiting publication in part or on whole as follows:

- **Chapters - 1, 2, 3, 4, 7**

  Shyam Reyal, Shumin Zhai, and Per Ola Kristensson. 2015.
  Performance and User Experience of Touchscreen and Gesture Keyboards in a Lab Setting and in the Wild. – ACM CHI 2015
  https://doi.org/10.1145/2702123.2702597

  In this paper, as first author, I myself carried out most of the work, which involves the study design, developing the experimental software required to run the studies, recruitment of participants, running the studies, followed by the analysis and presentation of the results. The co-authors helped in adding insight into the study design, and formulating the discussion and conclusions from this work.

- **Chapters 1, 2, 5, 6, 7**

  Shyam Reyal, Keith Vertanen, Per Ola Kristensson. 2018.
  Performance and User Experience of Typing and Speech on Mobile Devices in a Lab Setting and in the Wild - Manuscript

  As the paper above, I carried out a major part of the work as first author in this manuscript as well. I designed the study, developed the experimental software, recruited and ran participants, and followed with the analysis and presentation of the results. Co-author Keith Vertanen helped with preparing the phrase set with sentences containing varying degrees of perplexity and calculating the specific perplexity values. Co-author Per Ola Kristensson assisted with shaping the study design and writing the manuscript for publication from the thesis content.

1.9 Other contributions from this thesis

- **Contribution to Google Keyboard**

  As part of my summer research internship at Google Inc. in Mountain View, California, in 2015, I worked with the Google Keyboard team and contributed towards the (then) novel features Gesture Delete and Gesture Cursor Control, along with Dr Shumin Zhai (Senior Staff Research Scientist), and Kurt Partridge (Software Engineer). I contributed to the product by adding menu options to toggle the features on and off, the design and implementation of UI variations of Gesture Delete and Gesture Cursor Control, and running user studies internally among Google employees to compare these UI variations in terms of performance and user experience. The feature is now rolled out into production and comes as standard on all Google Keyboard releases on Android.

2

Literature Review

In this chapter, we provide a thorough survey of the related work in the field of mobile text entry, speech recognition, and evaluation methodologies for text entry on mobile devices. We do not delve deep into other areas of mobile text entry such as Handwriting Recognition – unless its evaluation methodology is of relevance - or novel text entry mechanisms such as Dasher (Ward, Blackwell, & MacKay, 2000), Speech Dasher (Keith Vertanen & MacKay, 2014) and their evaluations (Rough, Vertanen, & Kristensson, 2014) as they carry little relevance to the work carried out in this thesis.

We recognize and acknowledge previous surveys on mobile text entry, speech recognition and mobile text entry evaluation done by myself (Reyal, Zhai, & Kristensson, 2015), Kristensson, in his thesis chapter "Design Dimensions of Mobile Text Entry" (P.-O. Kristensson, 2007), Shabir et. al. in their "literature review on mobile devices touch screen inputs and its techniques evaluation" (Shabir, Tieng Wei, Abd. Ghani, & Kamaruddin, 2015), Kristensson, in his paper "the five challenges in text entry" (P. O. Kristensson, 2009), and (Mackenzie & Soukoreff, 2002).

2.1 Keyboard Based Mobile Text Entry

In this section, we explore the most relevant methodologies for the work presented in this thesis with relevance to mobile text entry using keyboards. These can be broadly categorised into physical keyboards and soft keyboards. Soft keyboards can be further divided into Keyboards that use Lexicons, Spatial Signals, and Language Models, Layout Optimised Soft Keyboards, Tilt Based Soft Keyboards, and Gesture Keyboards.

2.1.1 Physical Keyboards

Hardware keyboards are keyboard that are built into the hardware of the device. They have actual physical keys, and work by mechanical or electronic sensors. Some phone models include both the portrait and horizontal keyboards - portrait mode can be found on the device, horizontal mode can be found underneath the device, within an extendable compartment. The main benefit of physical keyboards are the touchable and perceptible user experience of having solid keys, which can be felt when pushed down and bounced back up, and clear key boundaries, which help in quickly making the typing embedded into muscle memory. (Shusterman, 2011)

Physical mini QWERTY keyboards are the smaller versions of desktop keyboards that are used in mobile devices. Unlike onscreen keypad, it provides proper tactile feedback, efficient text input (Cerney, Mila, & Hill, 2004). The BlackBerry^{TM} was a very popular type of device that incorporated a physical mini QWERTY keyboard which was meant to be operated using two fingers. Different people use different methods such as index finger, thumbs, single or both hands to operate the device conveniently in different environments. The two handed keyboard although smaller in size resembles the keypad of mobile devices. (James & Reischel, 2001)

Prior to the BlackBerry, the T9 and Multi-tap ("Software multi-tap input system and method," 2003) were very popular methods involving a physical, numeric keypad, following the ITU E1.161 standard for arranging letters and symbols on numerical keypads ("E.161 : Arrangement of digits, letters and symbols on telephones and other devices that can be used for gaining access to a telephone network," 2001). Around the same time, (Sirisena, 2002) in his experiments found that users can type at 9.23 WPM with 1.4884% error rates remaining on T9. (Sirisena, 2002) also proposed a new mechanism for typing on physical keyboards known as Fastap, which he benchmarked with T9 via a user study. Expert performance on Fastap was 8.86 WPM at a 1.26% error rate remaining.

(M. D. Dunlop & Crossan, 1999) presented a text entry method that was based on the T9 input mechanism. It is dictionary based and involves pressing a single button to shift characters - i.e. whether a ‘2’ is an “a”, “b” or “c”. This is carried out in context of a complete word with a dictionary as a reference for valid word. These were specialised for hardware keyboards with 12 keys and at most 5 function keys.

Hardware keyboards were replaced with software keyboards with time, as mobile phone design evolved from small screen with large keypads to devices with large screens that occupy the entire form factor – therefore the keyboard had to be part of the display itself.

2.1.2 Soft Keyboards

A software keyboard (or soft keyboard) has two main properties. The first is how it has been visualised – it looks like an actual keyboard but actually a rendered image on a touch-sensitive screen. The second is how it captures keystrokes – the aforesaid image has been set with a Touch Listener (in a platform independent mechanism) to listen for touch events on the specific points that graphically represent the keys on the keyboard. The literature has various nomenclatures for software keyboards i.e. virtual keyboards, graphical keyboards etc. We will use either of these terminologies interchangeably in the rest of this chapter.

Soft keyboards can be used with a pen or finger. However, since these do not have the same physical properties of hardware keyboards, the users do not enjoy the same tactile feedback and sensations as a physical keyboard. This loss of user-experience has three main aspects.

The first of these is that users do not feel a movement when a key is pressed i.e. a push down or push up, which means that the confirmation that a keystroke had occurred is lost. Software keyboards tend to work around this by providing a short vibration every time a key is touched. Using a stylus would mitigate this situation a little as the tip of stylus is a moving component that is “pushed down” when the tip touches a surface. (Brewster, Chohan, Brown, & Brown, 2007) showed that this mechanism was almost as fast as a physical keyboard.

Secondly, the touch might not register if the user does not apply the required amount of pressure, or, if their fingers are insulated. This sometimes is a problem when typing on touch-screens in cold countries, especially when the user has to wear gloves. This might also pose a problem in bad weather, when a mobile device (even waterproof ones) does not register a touch due the finger hydroplaning on the screen. This can be easily verified through personal experience. In contrast, the probability that a hardware keyboard fails to register a touch is very minimal.

Third and last, key boundaries no longer exist with soft keyboards. With a hardware keyboard, the key boundaries are significant, and a use can feel they are in the middle of a key, or have accidently moved between the boundaries or touched a surrounding key by mistake. With touch screens this is no longer the case. Mobile IME's try to work around this issue by providing key-press popups, where a key is zoomed and superimposed (Weir et al., 2014) (Baudisch & Chu, 2006) – with or without an offset (Weir, Rogers, Murray-Smith, & Löchtefeld, 2012) – upon keypress, but not with the same effect – as having these popups do not stop a user from accidently registering an unintended keystroke as seen in (Roudaut, Huot, & Lecolinet, 2008) and (Vogel & Baudisch, 2007).

Given these problems, software keyboards cannot stand on their own. To be accepted by the wider user population, and to be qualified as a "successor" to hardware keyboards, they must be complemented with error correction or prediction mechanisms. Work that has been carried out in terms of error correcting and predictive software keyboards can be found below.

2.1.3 Soft Keyboards that use Lexicons, Spatial Signals, and Language Models

(Goodman, Venolia, Steury, & Parker, 2002) proposed that a software keyboard can automatically correct typing errors using a character based language model, combined with a probabilistic language model made by the touch points. (Goodman et al., 2002) also found out that users achieved the same level of entry rate performance (20 vs 19.8 WPM) without correction and after correction. In this study participants were not allowed to correct errors, and the uncorrected error rate compared to standard desktop keyboard was reduced by roughly 1.8 fold. The system was evaluated in a short study with eight participants.

(P.-O. Kristensson & Zhai, 2005), proposed a geometric pattern matching technique that could also be used to correct typing mistakes on stylus keyboards. A limited performance study involving only the two authors, revealed that that they were able to achieve around 51 WPM in this study.

(Bi et al., 2014), in their paper “Both complete and correct?: multi-objective optimization of touchscreen keyboard” demonstrated that it is possible to simultaneously optimize a keyboard algorithm for both correction and completion. Correcting erroneous input and performing “auto-complete” on a partially entered word are two different aspects of improving accuracy when typing. This work showed that these features would complement each other, rather can cause a conflict. The experiments were conducted offline with no live participation from users.

(Weir et al., 2014), in their work titled “Uncertain Text Entry on Mobile Devices” explored the performance of two different touch models when entering text on touchscreen keyboards – a Gaussian model and a user-controlled touch model where the user can “influence” the uncertainty of via touch pressure. To evaluate the Gaussian model, they recruited 10 intermediate smartphone users and collected data over three 45 minute sessions to build the touch models. They further recruited another 10 participants – 8 of whom had intermediate smartphone experience – and compared the results with the state-of-the-art SWIFTKEY (“SwiftKey,” n.d.). They recruited another a 16 participants to evaluate the other mechanism “ForceType” as they called it, using a phrase set that contained slang and shorthand (Chen & Kan, 2012). They were able to show that with pressure adaptation, users were able to reduce the Active Correction Rate (ACR) from 19.48% to 10.86%. Further, with pressure adaptation they could type at 19.23 WPM as opposed to 15.42 WPM without.

(K. Vertanen, Memmi, Emge, Reyal, & Kristensson, 2015), presented VelociTap - a touch screen, onscreen keyboard decoder that uses sentence-based text entry. It significantly speeds entry rates by presenting three-word delimiter actions; a user can push the space button, swipe right or omit space-button and make the decoder infer by itself. When tested against Google's keyboard on Android devices, VeloicTap performed better by having significantly less error rate and using space key gave the most accurate results. If word-delimiter is made flexible, the error rate does not increase. Experiments revealed that novice users had on average 41 WPM rate and 3% character error rate.

(P. O. Kristensson & Zhai, 2008) described an adaptive lexicon method that splits a large lexicon into passive and active sets. The active set is slowly expanded to adapt to the users vocabulary (via lightweight interactions during active use). User studies and informal tests show that, on email texts input, the active set grew from 94%-97% of words entered by user to 98.5-99.0% of the words entered. It was found that recognition accuracy decreases as a function of lexicon size.

2.1.4 Layout Optimised Soft Keyboards

QWERTY is the standard and widely accepted layout for software keyboards in the market. (Shumin Zhai, Sue, & Accot, 2002) estimated the average expert text entry rate of QWERTY to be 34.2 wpm. (MacKenzie & Zhang, 1999) found that with a 20-minute transcription task users typed 28 wpm and had an uncorrected error rate of 3.2% on average. After 20 such sessions of 20-minute typing the average entry rate rose to 40 wpm with an uncorrected error rate of 4.8% remaining. Participants were not allowed to correct errors in this experiment.

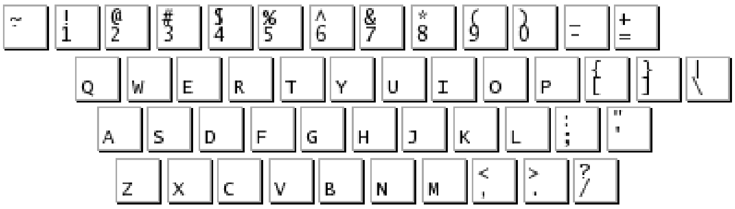


Figure 6 - QWERTY Keyboard Layout - https://en.wikipedia.org/wiki/Keyboard_layout

(Curran, Woods, & Riordan, 2006) stated that the international acceptance and the extensive use of QWERTY on a wide range of devices is unreasonable. It is a long standing understanding and observation that QWERTY layout is suboptimal (Levine & Goodenough-Trepagnier, 1990) especially with a single point of contact – i.e. a single finger, single thumb or single pen usage. When operating with two or more contact points e.g. all 10 fingers or two-thumbs – QWERTY keyboard becomes suboptimal but still good enough for practical use. It must be noted that QWERTY was designed to minimize mechanical jamming of the keys in typewriters (Hiraga & Ono, 1980). Therefore, mechanical arms corresponding to the most common pairs of keys were separated to opposite sides of the keyboard.

Researchers have tried to come up with different optimized layouts for the keyboard. (Lewis, LaLomia, & Kennedy, 1999) and (Helander, Landauer, & Prabhu, 1997) use a model based on Fitts' law (Fitts, 1954) and character-level bigram statistics to find more efficient keyboard layouts.

(Mackenzie, Zhang, & Soukoreff, 1999) identified and explored the novice and expert text entry rates of QWERTY, ABC, Dvorak, Fitaly, JustType, and telephone keyboards. 24 subjects were used to test them. The results showed that novice participants had 8-10 WPM rate due to time taken in finding the right keys, while expert participants had WPM rate of 22-56. In a novice test, participants achieved 20.2 WPM for QWERTY, 10.7 WPM for ABC, 8.5 WPM for Dvorak, 8.0 WPM for Fitaly, 7.0 WPM for JustType and 8.0 WPM for telephone, with the QWERTY WPM rate consistent with other studies and which also suggests that there is a skill transfer from computers to stylus-based tapping devices.

(MacKenzie & Zhang, 1999) then evaluated a high performance soft keyboard called "OPTI" which they empirically found out to be 35% faster than QWERTY. Their study participants initially typed 17 wpm with OPTI and had an average uncorrected error rate of 2.1%. After 20 x 20 minute sessions the average entry rate rose to 45 wpm with an uncorrected error rate of 4.2%.

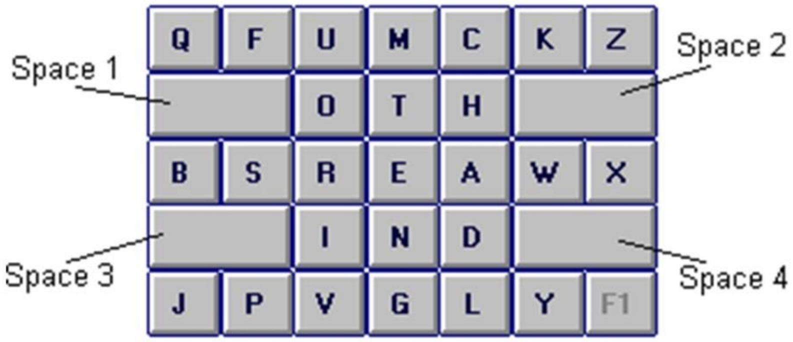


Figure 7 - OPTI Keyboard Layout - https://www.yorku.ca/mack/CHI99a.html

To show that device independent text entry is possible, a device independent text entry mechanism was proposed by (Isokoski & Raisamo, 2000) - which allows to transfer skills across devices. This was done by using those characteristic of text entry, which are common across current devices - via the Minimal Device Independent Text Input Method (MDITIM), which allows word level uni-strokes hand-writing. Results show that it is likely that MIDTIM is not fast enough to compete with current fast device dependent methods.

(Shumin Zhai et al., 2002) carried out a study on Movement Model, Hits Distribution and Learning in Virtual Keyboarding where they compared how participants performed in different optimized keyboard layouts. Both the above studies were carried out using stylus-based input. Further, (Shumin Zhai et al., 2002) estimated the average expert entry rate performance of "OPTI" to be 42.8 wpm.

(S Zhai, Hunter, & Smith, 2002) proposed a new layout METROPOLIS, which was developed using a metropolis random walk algorithm. For this, they didn't rely on heuristics or trial and error approaches. In their paper, they found the average expert text entry rate is 46.6 WPM.

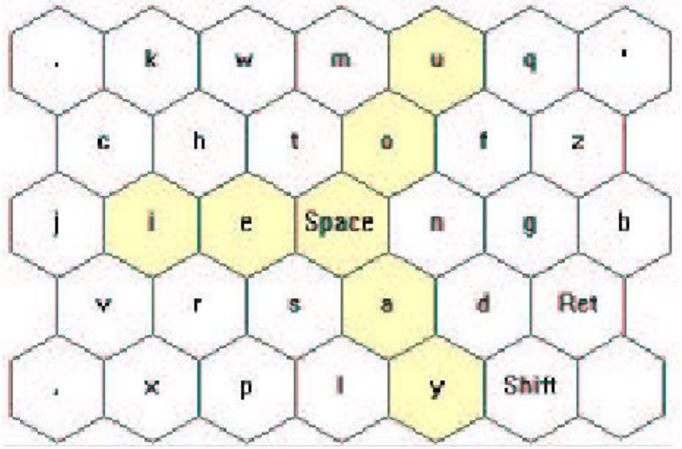
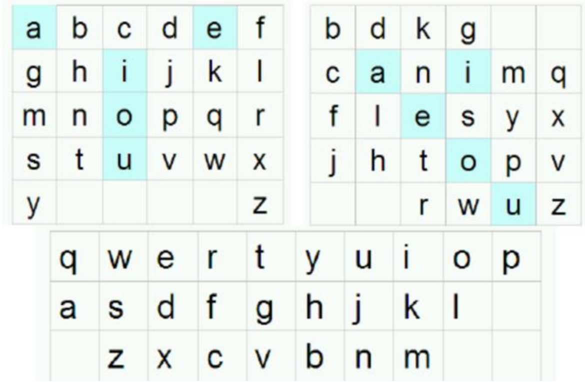


Figure 8 - Metropolis Keyboard Layout

https://www.researchgate.net/publication/2487770_Movement_Model_Hits_Distribution_and_Learning_in_Virtual_Keyboarding/

Another example for algorithmically derived keyboard layout is ATOMIK by (S Zhai et al., 2002), which uses a Fitt's Law based movement model, a letter bigram frequency model, and alphabetical ordering. They estimated the average expert entry rate to be 45.3 WPM. However, the entry rate seems to be reduced than of the standard. (Smith & Zhai, 2002) in their user study suggest that having alphabetic ordering is found easier to type by novice users than without.



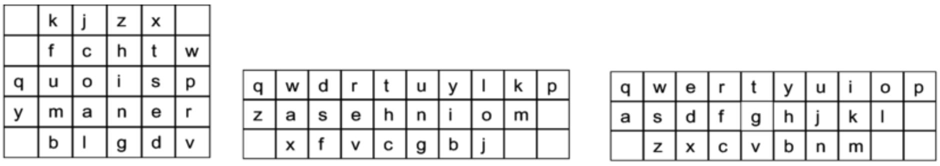
a	b	c	d	e	f
g	h	i	j	k	l
m	n	o	p	q	r
s	t	u	v	w	x
y					z

b	d	k	g		
c	a	n	i	m	q
f	l	e	s	y	x
j	h	t	o	p	v
		r	w	u	z

q	w	e	r	t	y	u	i	o	p
a	s	d	f	g	h	j	k	l	
	z	x	c	v	b	n	m		

Figure 9 – ATOMIK and Interlaced QWERTY Layouts (Shumin Zhai & Kristensson, 2008)

(Bi, Smith, & Zhai, 2010) explored what they called "Quasi-Qwerty Optimization" which resulted in optimized close-to-Qwerty keyboard layouts. This was particularly interesting in terms of usability as the designers tried to keep the design close to QWERTY, thus making it easier and more adaptable for first time users to pick it up and adapt easily to the new layout.



	k	j	z	x	
	f	c	h	t	w
q	u	o	i	s	p
y	m	a	n	e	r
	b	l	g	d	v

q	w	d	r	t	u	y	l	k	p
z	a	s	e	h	n	i	o	m	
	x	f	v	c	g	b	j		

q	w	e	r	t	y	u	i	o	p
a	s	d	f	g	h	j	k	l	
	z	x	c	v	b	n	m		

Figure 10 - quasi QWERTY and QWERTY layouts (Bi et al., 2010)

(M. Dunlop & Levine, 2012) presented two new touchscreen keyboards based on Pareto optimization techniques. The user study consisted of four short trial sessions where the difference in entry rate between the new layouts and a regular QWERTY were not significant.

(Oulasvirta et al., 2013) presented KALQ, a split keyboard layout optimized for two-thumb typing that assigns letters to keys computationally in order to minimize travel distance and maximize the alternation between the thumbs. KALQ layout is designed due to the way users grip the device with two hands. After eight hours of practice, six study participants eventually typed faster using KALQ than a QWERTY baseline - with users reaching 37 wpm (5% error rate) after being trained.

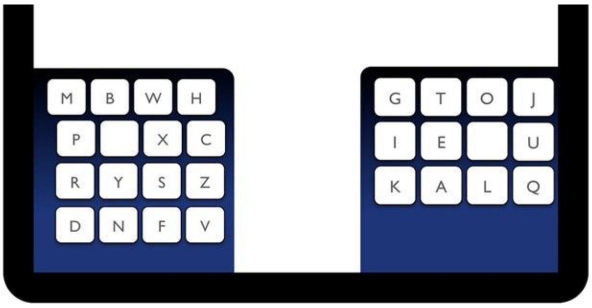


Figure 11 - KALQ layout - https://newatlas.com/kalq-keyboard-touchscreens/27140/

The resistance when using optimised keyboard layouts is that users have to learn a completely new layout. (Shumin Zhai et al., 2002) found that learning rate can be improved by expanding the rehearsal interval of users. Further, (Bi & Zhai, 2016) found out that by changing one key, swapping the key positions of I and J in the QWERTY layout, results in a layout that is extremely easy to learn. Their research further showed that disparity from QWERTY substantially affected the learning of a new layout and to minimize such efforts the new layout should have a strong resemblance to QWERTY.

Another novel approach by (Magnien, Bouraoui, & Vigouroux, 2004) was that the system highlights the keys which it calculated that are more likely to occur, based on what the user has already typed. They hoped that this would reduce search time when practicing a new keyboard layout. In their paper, they show that input rates rise when keys are highlighted for novice users – who entered around 50 words with and without prediction. The users were kept novice by rearranging the keys before each word, however, how the exact study design (i.e. mixed design, counter balanced conditions etc.) were missing. However this can have practical applications when teaching new users to type or helping users with special needs.

Nel, MacKay and Kristensson (Nel, Kristensson, & MacKay, 2018) presented a novel stereophonic and probabilistic single-switch text entry method, which is useful for visually impaired users with motor disabilities. These users would usually rely on single-switch scanning systems for chatting. In Ticker, model based interaction with statistical models for robustness allows inference of text in presence of noise. The approach was evaluated via simulations and user studies.

2.1.5 Tilt Based Soft Keyboards

Researchers have used the mobile device's sensor information to grasp device tilt and use it to optimise the text entry experience by supplementing keyboard typing. In TiltType, as presented by (Partridge, Chatterjee, Sazawal, Borriello, & Want, 2002) the user can press special buttons and tilt the device in order to get the best disambiguation of what is typed.

Further, (Wigdor & Balakrishnan, 2002) used tilt information to further decode and disambiguate keypad presses. The user can use the four directions to disambiguate from four possible letters for a given key – i.e. tilt left for 2 on the 2ABC key, tilt forward for A on the 2ABC key and so on. However, the resultant entry rates were slow. Participants could only reach an average 7.53 WPM after 20 minutes of practice, and 13.57 WPM after 250 minutes of practice. They baseline condition here was multitap, and users were required to correct all errors to continue.

(Yeo, Phang, Castellucci, Kristensson, & Quigley, 2017) proposed and evaluated a tilt-based text entry method for single handed usage. They had taken influence from a gesture keyboard, but instead of drawing gestures - gestures are produced by tilting the device. The experiment included 12 participants for a transcription task and 6 participants for composition. Novice users showed to have speeds of 15 wpm after some practice and a single participant achieved 32 wpm after 90 mins of user. The paper advised that tilt-based gesture text entry be used as an alternative to single-handed typing.

2.1.6 Gesture Keyboards

(Perlin & Ken, 1998), presented QuikWrite - a stylus based text entry method in which the stylus is never lifted from the surface and the user does not require to stop the motion of the stylus. Hence, multi-word text of any length can be entered fluidly. Implemented as a Java applet and as PalmOS app, the user is presented with a simple alphabet with the writing areas divided into zones arranged in a 3x3 grid. Characters are formed by dragging the stylus into these zones and gestures are implemented for frequent characters (and for ‘space’). Other gestures can be used to change case. When tested against ‘Graffiti’, users found Quikwriting to be 3x as fast with very high accuracy. Users required 2-3 hours of practice before using it confidently.

(Shumin Zhai & Kristensson, 2003) presented SHARK - which was later refined into what was then called the SHARK^2 system (P.-O. Kristensson & Zhai, 2004), which eliminated the requirement to alternate between the two methods of entry - gesturing or tapping, thus allowing any word to be entered as a shorthand gesture. The evaluation of the SHARK^2 system used only two participants (the two authors) and a few short trial runs, and reported only on the “record entry rates” based on an optimized stylus keyboard layout.

(P. O. Kristensson & Zhai, 2007) expanded the Gesture Keyboard paradigm to also enable users to issue commands. This system used shorthand gestures to enter commands (for example, *Cut*, *Copy*).

Gesture Keyboards have also been studied by Kristensson (P.-O. Kristensson, 2007), who conducted two studies comparing the performance of a gesture keyboard on a pen-based Tablet PC with a physical two-thumb keyboard. However, the first study only investigated the first 25 minutes of writing performance of the SGK, and the second study used a "motor memory saturation" setup in which participants repeatedly wrote the same phrases over and over. Kristensson (P.-O. Kristensson, 2007) found that the SGK in his study resulted in 25 WPM on average after 25 minutes of writing. In the second study, participants wrote text at 40 WPM on average.

(Rick & Jochen, 2010) investigated the influence of the keyboard layout on expert text entry performance for the gesture keyboard. Almost a year later, (P.-O. Kristensson & Vertanen, 2010) combined gesture input and speech recognition, allowing users to flexibly enter text using both methods of input in order to reduce error rates.

(Castellucci & MacKenzie, 2011) presented another brief study of the SGK in conjunction to their presentation of a text entry experiment tool for Android devices. Their study involved six participants using an SGK for 20 minutes. The final entry rate for SGK was 20 WPM. Another study was presented by Nguyen and Bartha (Nguyen & Bartha, 2012) conducted another small study on the performance of Gesture Keyboard on a Windows 7 tablet. Their study involved 14 participants using the SGK for five minutes. The entry rate for the SGK was 12 WPM.

(Bi et al., 2012) presented the bi-manual gesture keyboard that enables users to use both thumbs to write two gestures that represent a single word simultaneously on a split keyboard, breaking the single-handed nature of gesture keyboards. However, a user study did not demonstrate any performance improvement with the bi-manual gesture keyboard compared to the traditional uni-manual gesture keyboard.

Empirical studies of the performance, or experience, of writing using gesture keyboards are generally difficult to conduct because text entry is a complex form of interaction involving motor skills, memory, learning and other cognitive aspects of human behaviour. These factors may change from the laboratory to real world everyday use environments. (Reyal et al., 2015) The age-old debate of the QWERTY vs. the DVORAK keyboard illustrates the lack of power of simple and relatively short empirical studies that do not saturate users' learning. (Lewis et al., 1999). In the context of modern Smart Touch Keyboards (STKs), the topic is even more complicated because inevitably the empirical results are to a large degree dependent on the algorithms, parameters, and the sizes of the keyboard vocabulary, and product design in general at the time of the study.

What we see from the above literature is that the empirical research is limited on gesture keyboards and mostly the studies involve inductive stylus touchscreens, focusing on individual aspects of the gesture input paradigm, such as gesture memorability, ceiling performance, and impact of the layout.

2.2 Speech Based Mobile Text Entry

In this section, we explore the most relevant work with regards to speech recognition on mobile devices.

Even though human beings can speak around 200 WPM (Rosenbaum, 1991), it doesn't mean that a computer system can interpret a human beings verbally spoken messages into text (Moore, 2004). Karat, in her paper "Patterns of entry and correction in large vocabulary continuous speech recognition systems" (Karat, Halverson, Horn, & Karat, 1999), identified some vital statistics such as humans can effectively transcribe text at a speed of 13.6 WPM, when instructed to correct text ensuring no errors.

(Shneiderman, 2000) argued that speech is better off as a command interface rather than a text input interface, given speech could drain cognitive resources and users may find the experience frustrating. Also natural speech tends to be conversational and informal, therefore as (Shneiderman, 2000) argues, an effective human to human communication method – i.e. speech, does not mean an effective human to computer interface.

(H Fischer, Price, Sears, & Price Andrew Sears, 2005), in their work, argue that speech decoding must be off-loaded to an external server, due to the limited computational power on devices. This is even in practice today with most state of the art speech recognition engines, such as Google Speech Engine, Baidu Speech Engine and Amazon Alexa. They therefore introduced network latency as a factor affecting entry rate.

(Shneiderman, 2000) summarized that as the number of speech interaction methods increase, there are more empirical studies being carried out on their success and limitations. Particularly, there is a concern for drivers and their safety as manufacturers plan to add speech-to-email writing facilities. There is a fear of increased accident rates. Hence, we need realistic goals for human-computer interaction based on speech. Components of human-human relationships, such as friendships and inspiration are not associative to human-computer interaction. Among these concerns, speech-recognition is still beneficial for blind and limited-mobility users. As with physical devices, control parameters will be effective, while keeping noise levels low for human-human chat.

(Petajan, Bischoff, Bodoff, & Brooke, 1988), the authors of "An improved automatic lip-reading system to enhance speech recognition" suggest that since current speech recognition technologies perform well with small vocabularies in noise or with large vocabularies in high noise, speech recognition in noise can be improved via automatic lip reading. They modify a previous technique which now uses dynamic time warping, vector quantization and heuristic distance measure to give visual speech recognition results from multiple speakers. The experimental methods include image processing, nostril tracking (for locating the mouth), and visual word capture (via a contour coder, which stored 200 video frames and then vector quantization of mouth images. 4 speakers were chosen for this experiment.

(Bassil & Alwani, 2012), the authors of "Post-Editing Error Correction Algorithm For Speech Recognition using Bing Spelling Suggestion" identify that Automatic Speech Recognition (ASR) is error prone and imprecise if used in a noisy environment. A post-ASR editing method based on Bing's spelling suggestions is designed, in which the ASR recognized output is send to Bing for spell checking and correction. The method breaks down the ASR output into tokens that are sent as search queries to Bing. Experiments that involved various speeches in different languages that the number of ASR errors decreased with this approach.

(P.-O. Kristensson & Vertanen, 2010), the authors of “Asynchronous Multimodal Text Entry using Speech and Gesture Keyboards” describe a merge model which aims to reduce text entry errors by combination of speech and gesture keyboard. The results are combined in a flexible and asynchronous manner. Experimentation of this method included the collection go gesture and speech data from participants of this experiment, in which the users had to enter short email sentences and web search queries. The word rate error got reduced by 53% and 29% for email sentences and web-searches respectively. When the user did not indicate wrong words, the model reduced the word rate error by 44 %.

(Keith Vertanen & Kristensson, 2010b), explored ways of improving recognition in their work “Getting it right the second time: recognition of spoken corrections” Simple techniques, such as silence filtering reduced the error-rate rate from 54.9% to 30.8%. A flexible model is devised which combines information from spoken correction and the original recognition to improve accuracy. This is done by using confusion networks and prior beliefs about the recognition events. With all these techniques, accuracy (correctly input words) increased to 53% from 21%.

(Keith Vertanen & Kristensson, 2010a) described that as correction of errors in recognition are an important step, ways to better support this via the Parakeet software were explored. Parakeet is a continuous speech recognition system for mobile devices which allows easy correction on these devices. From a confusion matrix, users are made to select alternate words (while typing on a predictive text keyboard). Experiments with participants showed that although sometimes the initial recognition error rates were high, the users were still able to write effectively.

(Keith Vertanen, Vertanen, & Kristensson, n.d.) Investigated ways to correct voice web search queries. A corpus of web-search queries is described - which helped to show that search-specific vocabulary (pronunciations are generated automatically) is better than vocabulary based on fixed pronunciations. Experimentation shows that even though the word error rate was 48%, participants only took on average 18 seconds to search and correct their search queries.

(Ruan, Wobbrock, Liou, Ng, & Landay, 2016) compared touch screen keyboard and speech-based dictation methods for English and Mandarin. For speech recognition methods, an initial transcription was provided, and then recognition errors could be corrected by using speech again or shifting to keyboard. With speech, the English rate was 3 times faster and the Mandarin rate was 2.8X faster than using keyboard. Furthermore, the error rate for English was 20.4 % lower and for Mandarin, the error rate was 63.4 % lower than keyboard. Deep Speech 2, was used to conduct this experiment, which had built-in QWERTY and PINYIN iOS keyboards. Results show that there is a significant shift from typing to speech.

2.3 Evaluation Methods for Mobile Text Entry

In this section, we explore the most relevant methodologies for evaluating mobile text entry mechanisms. Broadly, we can divide the evaluation work carried out so far into simple A|B comparison studies, and unconstrained studies. We also explore related work on software apparatus – such as tools and phrase sets – provided for the study of mobile text entry.

We also reviewed existing research publications on how the experience sampling method has been used in studies. An interesting contribution is how to use ESM to evaluate Ubicomp applications by (Consolvo & Walker, 2003) of Intel Research.

2.3.1 A|B Comparison Studies

These studies are mostly conducted in a lab environment, with the user being seated, with one or many controlled variables, with all the other conditions controlled in order to minimise cofounding factors. These are normally used to compare a novel (proposed, new) text input method with a baseline.

(Keith Vertanen, Memmi, & Kristensson, 2013) suggested that as typing on a touch screen is very difficult without actually looking at the screen, a new approach is presented where a keyboard is imagined on any part of the screen and users type out the whole sentence without any visual response. To demonstrate this, a decoder is developed which consists of a keyboard topology mode, tap variability and a statistical language model. Experimentation shows that novice users with highly noisy input have one-third of sentences decoded without errors.

(P. O. Kristensson & Denby, 2009) of University of Cambridge, report on the study on performance of unconstrained handwriting recognition. 12 participants from different departments were chosen (and screened for dyslexia and RSI). 7 of these were native English speakers and 5 had English as second language. Using a Dell XT Tablet PC, participants had to use a touch-pen to perform handwriting recognition tasks, where the recognizer adapted individually to each participant. After 250 minutes of practice, they had a mean text entry rate of 24.1 words per minute - and for the first 4 hours of the experiment, the entry/error-rates were same as a baseline QWERTY keyboard. In total 100 hours of data was collected, during which the participants entered on average 83.2 phrases with a software keyboard and 81.5 phrases with handwriting recognition. This shows that handwriting recognition performs at part with a QWERTY keyboard.

(Keith Vertanen, Fletcher, Gaines, Gould, & Kristensson, 2018) suggested that, as on-screen keyboards allow us to one word at a time, user performance and recognition accuracy needs to be compared in terms of phrase-at-a-time, sentence-at-a-time and word-at-a-time entry on a smartwatch keyboard. After experimentation, the results suggest that when entire sentence is input, the accuracy increases from 26 wpm to 32 wpm with character errors less than 4%. These findings then suggest that virtual keyboards can enhance performance by allowing users to enter more information at a time.

(Clawson, Lyons, Starner, & Clarkson, 2005) investigated blind typing on mini-QWERTY keyboards by studying 8 users on 5 typing sessions (each of 23 minutes). These were done with and without visual feedback of keyboard and/or the screen. The participants had a mean WPM of 45.8 at 85.6% accuracy. This was compared with Twiddler, on which the WPM and accuracy rose slightly.

Lastly, two interesting studies have been conducted on how touchscreen keyboards fare during various situations. The first of is WalkType (Goel, Findlater, & Wobbrock, 2012) which is a touchscreen keyboard correction algorithm designed to support people walking, and ContextType (Goel et al., 2013) which uses hand posture information to improve the mobile text input experience, both presented by Goel et al in 2012 and 2013 respectively.

2.3.2 Unconstrained Studies

These studies are mainly conducted with the aim of revealing more information about a certain text entry mechanism or a group of mechanisms in more unconstrained settings – such as outside of the lab – or as we call it, in the Wild, or via crowdsourcing.

(Wilson, Brewster, Halvey, Crossan, & Stewart, 2011) investigated how walking impacted linear pointing performance on mobile devices. They observed lower response times and higher error rates when participants were walking as opposed to being seated. This work suggested that mobility severely affected pointing tasks in terms of error rate.

(Henze, Rukzio, & Boll, 2012) in their paper “Observational and experimental investigation of typing behaviour using virtual keyboards for mobile devices” presented a typing game that records users behaviour when they touch the standard android keyboard. There were around 73K installations which gave over 47M keystrokes. In the second part of the work, they proposed to visualise the touch points on the keys using a dot using a “shift” function which used the data generated from the first part of the study that reduced the error rate by 18.3% and improved performance by 5.2% with no effect on learnability. This was observed via 6M keystrokes obtained across 13K installations.

(Dhakal, Feit, Kristensson, & Oulasvirta, 2018) studied and reported the users’ behaviour in keeping with their performance. A total of 168,000 volunteers were used.

Hence a comprehensive database is established which allowed statistical analysis in terms of linking of keystroke patterns to typing performance. The experiment reports that when letter pairs are typed with difference hands/fingers, they are more predictive in typing speed; and that the issue of roll-over typing is very common. Roll-over typing refers to the technique where the keyboard layout is such that two consecutive keypresses happen with two different hands. Therefore, when parallelizing the finger movements a.k.a. one finger lifts off while the other is moving down for the keypress, this significantly reduces the time it takes to enter text than when not using roll-over typing. The authors suggest that users can be divided into 8 groups based on their performance, accuracy, rollover and hand/finger usage.

2.4 Other Important Findings

We identify and acknowledge a few important findings in the text entry literature that we find relevant to the work presented in this thesis.

(P. O. Kristensson, 2009) identified five challenges of AI-based text entry – they were Localization (variety of keyboard layout based on country), Error correction (cognitive and motor errors), Editor support (AI-based text entry methods may output multiple recognition candidates, an editor should support probability of recognized word based on previous words), feedback (immediate or delayed feedback) and context of use (contexts that the users are exposed to in daily life).

(P. O. Kristensson, 2011), in his paper titled “Design dimensions of intelligent text entry tutors”, showed us that as AI based text entry methods require training process from users, the development of intelligent text entry tutors would reduce this training time investment. The paper then further shows us design dimensions of this approach, namely, automaticity, error correction, engagement and feedback.

(P. O. Kristensson & Vertanen, 2014) also presented the inviscid text entry rate - which is the point where text entry is bottlenecked by user’s creativity than the method itself. This concept is applied to find the grand goal for future mobile text entry methods. In the paper, it is estimated that on average the inviscid rate is 67 wpm and current mobile text entry methods have significantly slower. Hence as future work, methods need to be developed which can approach the inviscid entry rate.

2.4.1 Metrics

(MacKenzie, 2002) defined the calculation of KSPC and provides its examples for a variety of text entry methods. Key Strokes Per Character (KSPC) is simply the number of keystrokes the user entered to enter the text, normalised by the number of characters in the resulting text. I.e. the number of keystrokes can differ from the resultant number of characters as the user may correct or delete text to arrive at the result e.g. by the use of backspace, spacebar, and arrow keys. Experiments revealed that for a QWERTY keyboard, the KSPC is 1.0; Keypads, more than 1; Word prediction had the prediction of KSPC less than 1 as words can be added without typing all characters. It expected that there is a inverse relationship between KSPC and throughput, though other factors which affected KSPC were repeat keystrokes and attention demand for the particular method.

Further, (Soukoreff & MacKenzie, 2003) identified that there are shortcomings in evaluating text-entry using minimum string distance (MSD), and keystrokes per character (KSPC). The minimum string distance (MSD) between two strings is the minimum number of insertions, deletions, or substitutions that are needed to transform one string into the other.

Hence, a new framework is created for error analysis which combines analysis of the text, input stream/keystrokes and the transcription. The framework provides measures such as unified error-rate, error correction efficiency, and utilized bandwidth among others. These new error rates reflect all errors by a user (corrected or not). Furthermore, corrected and not corrected errors are noted separately, with the error rates being device independent. Moreover, it was seen that when the defined text is hidden as the user starts to enter text, the text entry speed increased with a higher not corrected error rate. The authors wished to extend it to non-keyboard based methods.

2.4.2 Phrase Sets

(MacKenzie & Soukoreff, 2003) proposed “phrase sets for evaluating text entry techniques” which has been cited in 256 published works so far (as at 01/10/2018), which formed the basis for using standardized phrase sets for text entry experiments.

(Keith Vertanen & Kristensson, 2011) argued that, at the time (2011), there was a lack of a phrase set composed for mobile phone users. Due to this reason researchers create their own phrase sets - which may be unknowingly biased and ineffective in gauging accurate error rates. Hence, a collection of actual email sentences written on mobile devices was collected (Blackberry phones). It was then empirically established that the sentences were east to memorize and the accuracy and speed with these sentences could be typed with a full-size keyboard.

(Kano, Read, & Dix, 2006) investigated the suitability of current phrase sets available in HCI for the use of children. They suggested that the current phrase sets may be unsuitable for children and proposed a new phrase set containing 500 phrases taken from children’s books, which was experimentally shown to be more suitable by having 40 children between the ages of 7 and 10 evaluate the phrase set using 4 identical black keyboards. This study also revealed how children perform copy tasks, perform errors and how they carry out corrections.

(Kano & He, n.d.) evaluated phrase sets for use with text entry methods for dyslexic participants. In their work, the authors identified that additional evaluations should be carried out for dyslexic participants in addition to standard text entry evaluations, and analysed to what degree a given phrase set as certain “trigger words” that affect dyslexic people. It was shown that the two phrase sets analysed in the paper (James & Reischel, 2001; MacKenzie & Soukoreff, 2003) had a high proportion of trigger words and the occurrence of these should be reduced to be successfully used with dyslexic participants.

These phrase sets were publicly evaluated by (P. O. Kristensson & Vertanen, 2012), in two large scale crowdsourced text entry experiments. They also evaluated the effect of memorization of phrases vs seeing the phrases for the first time during a transcription task. They studied the aspects reproducibility, heterogeneity, internal validity and external validity with respect to using each of these phrase sets and provided recommendations on their usage.

2.4.3 Transcription vs Composition Task

(Keith Vertanen & Kristensson, 2014) proposed the addition of a composition task to the text entry evaluations - providing higher internal validity for transcription task and higher external validity for composition task. The authors argued that the *de-facto* research methodology of using a transcription task has high internal validity yet low external validity. According to this work, the transcription task is advantageous such as (a) all participants write the same text – making the evaluation easier (b) reducing the cognitive load on the participant – as they do not have to think what to write before writing it (c) the participants can internalise the stimuli, memorise it, and then start to type, instead of thinking and typing at the same time. However, they also point out disadvantages such as users not having the same writing styles and preferences – i.e. not using the same words to convey the same message. Therefore adapting to a different style might provide an exaggerated higher or lower performance and user experience, by having higher cognitive effort to adapt to the task. However, the researchers showed that, large-scale, crowdsourced experiments revealed that the users would invent consistent and rapid creative and high-quality compositions with insignificant reduction in text entry rate.

(Keith Vertanen, Emge, Memmi, & Kristensson, 2014), in their paper titled “TextBlaster proposed a multiplayer game in which players use precision, speed and timing of their own typing to be the last person standing in this shoot ‘em up game. As a sentence-based decoding approach is used, the auto-correction infers after a whole sentence is typed. As the game is competitive, it forces the users to enter text carefully and quickly; which make Text Blaster ideal for running text entry experiments.

2.5 Concluding Remarks

From all the examples above, we see that the empirical studies reported of these research systems tended to be small in scope (of learning or in terms of the number of study participants, or both). Many of these studies are also based on stylus keyboards on relatively large inductive touchscreens using relatively simple research prototype software, in contrast to product-ready systems.

Nonetheless, not having any in-depth empirical studies is not acceptable for the HCI field. Continued progress and innovation in the text entry field cannot have a solid empirical footing if we do not even know how well the current technologies work for users.

3

Study A – Comparison of STK and SGK in a Lab Setting

3.1 Motivation

Even though STKs and SGKs have become the mainstream touchscreen text entry methods, the HCI research literature offers little empirical evaluation of the current state of affairs in general, and the performance and experience difference between STKs and a SGKs in particular.

Empirical research has been limited in scope, size, and technology form factor. Most reported text entry research has also been based on research prototypes. Continued progress and innovation in the text entry field cannot have a solid empirical footing if we do not know how well current technologies work for users. Further, despite the prevalence of STKs and SGKs there is a lack of in-depth studies about their text entry performance, in particular outside a lab environment.

In this chapter, we empirically compare two state-of-the-art text input methods in a controlled lab environment. In the next chapter, we empirically investigate how the two input methods perform outside a lab environment.

3.2 Hypotheses

We present the following null hypotheses which are to be accepted or rejectd as a result of this study. As shown, this study is broad and sheds light on many different aspects of text input between the two keyboards.

H0,a There is no difference in text entry rate between the two keyboards (STK and SGK) in a lab setting

H0,b There is no difference in character error rate after correction between the two keyboards (STK and SGK) in a lab setting

H0,c The entry rate does not differ between different hand postures used, in a lab setting

- STK using single index finger
- STK using single thumb
- STK using two thumbs
- SGK using single finger
- SGK using single thumb

H0,d The error rate does not differ between the different hand postures used (see H0,c) in a lab setting

H0,e The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of entry rate, in a lab setting

H0,f The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of error rate, in a lab setting

H0,g The user experience of the participants did not differ between STK and SGK in a lab setting

3.3 Variables & Confounds

In this study, we identify three types of variables as independent variables, dependent variables, and confounds. Some of these have been defined and described in Chapter 1 – Introduction, what is described here are the ones specific to this study.

3.3.1 Independent Variables

These are the variables we explicitly control in this study.

- V1 – Keyboard Type (2 levels: STK, SGK)
- V2 – Participant (12 levels: P1-P12)
- V3 – Session (5 levels: S1-S5)

3.3.2 Dependent Variables

These are the variables that we measure as an outcome of this study. The measurements lead to "derived dependent variables" which lead to the analysis of the study results. This means we do not measure these directly but we derive them via calculations from the dependent variables we measure. The following sub sections below describe the variables we measure vs the variables we derive.

3.3.2.1 Measured Dependent Variables

These are the dependent variables we explicitly measure.

3.3.2.1.1 Timestamp at first keystroke (T1)

Theoretically, this is when the first key is touched as the user begins to type. However, with proprietary keyboards such as GBoard, we are not provided with a call-back function when a key is entered or a gesture trail begins as the user starts to type or gesture. Therefore we use a practical alternative – in both typing and gesturing, the user first has to touch the target text field (a *TextView* in the case of Android application), to bring it to focus. This would bring up the soft keyboard and fire an *onKeyDown* event, which we can capture. Although this is not exactly when the user starts to type but slightly earlier we believe this is a good estimate of when the user begins to type as (a) the users begin to type immediately after the keyboard comes up (b) we ask the users to first "internalise" (or memorise) the stimulus phrase and then bring up the keyboard.

3.3.2.1.2 Timestamp at final keystroke (T2)

Theoretically, this is the timestamp when the user enters the last character in the sentence or phrase they intend to type. This can be captured with regards to typing on an STK by using the last *onKeyUp* event on the *TextView*, in the case of the Android platform. However, this is impossible to capture in the case of Speech input, as the speech capture interface continues to run after the user has stopped speaking, waiting for a significant spell of silence before it deactivates. Therefore when running studies, we use a practical delimiter to capture when the user has completed typing – such as pressing a button which says NEXT, or FINISHED. Realistically, in a texting mobile application this would be denoted by pressing SEND. In this study, we capture the "end of phrase" when the user indicates they want to move to the next sentence by pressing NEXT.

It is obvious that there is a slight delay between entering the last character in the response phrase and pressing next, however, this does not skew the results in the study as:

a. When typing continuously, this happens almost instantaneously
b. We explicitly tell the users to use a minimal delay between finishing typing and pressing next
c. This delay is uniform across the entire study (and does not differ much between subjects)
d. If the user does require to proofread what they typed, this should be indeed factored in to the time it takes to enter text using the given input mechanism, as this is a critical factor

3.3.2.2 Derived Dependent Variables

These are the dependent variables we calculate from the measured dependent variables. Descriptions of these can be found in Chapter 1 – Introduction.

- Number of characters in the response phrase (N)
- Typing duration (T)
- The Error (E)
- Entry Rate (WPM)
- Error Rate

3.3.3 Confounding Variables

These are variables that we did not try to control, but still would be consider as variables due to their confounding nature, as they can definitely affect the typing experience and performance in the study. Descriptions of these can be found in Chapter 1 – Introduction.

- OOV words
- Hand Posture

3.4 Apparatus

3.4.1 Hardware

We used two identical LG Nexus 4 mobile devices running Android 4.3. The 4.7" Corning Gorilla Glass 2 touch screen had a resolution of 1280 × 768 pixels at 320 pixels per inch. The physical devices measured 133.9 × 68.7 × 9.1 mm.

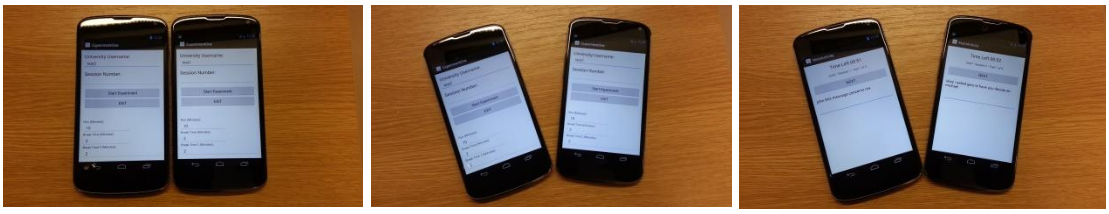


Figure 12 - Hardware apparatus used for the study

When these studies were run, the Nexus 4 was the best representative device for stock Android, which was the main reason behind using this particular brand and model. Full hardware specifications can be found at (AndroidCentral, 2012)

3.4.2 Software Apparatus

There were two major components in the software apparatus. The First was the Google Keyboard, which had its own implementation of state-of-the-art STK and a state-of-the-art SGK built in. The second was the experimental software that was required to run the study.

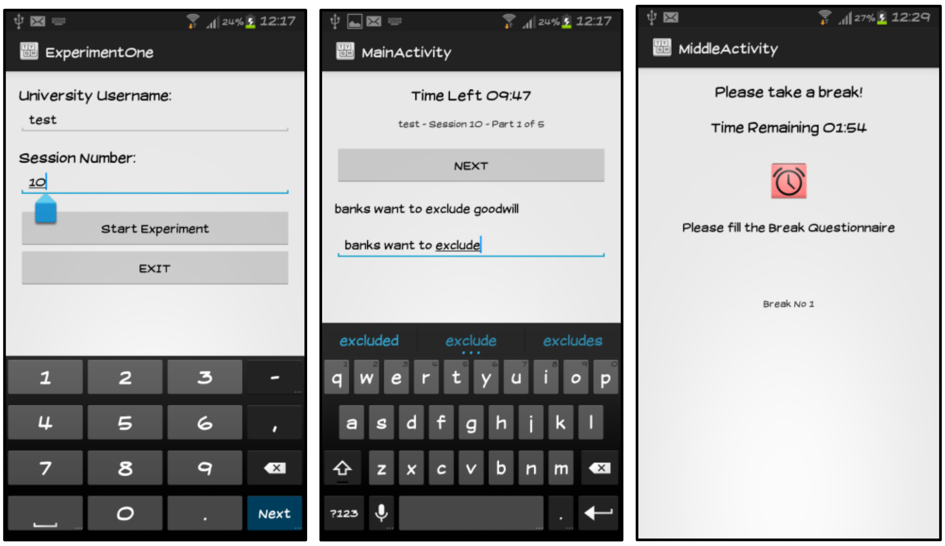
3.4.2.1 Experimental Software

The app used for the study outlined in this chapter is designed for a typical longitudinal study carried out in a lab environment, where a participant would repeatedly enter response phrases to a stimuli phrase shown to them, for a set duration, with breaks in between. The app will record the stimuli and response phrases, and the elapsed time, which could be used to analyse results.

The specialty of this app is that it allows the fully automatic execution of the experiment without the intervention of the researcher. The app assumes two conditions, and allows the researcher to specify which condition to use for each experiment session. The app provides an interface to provide configuration parameters for the experiment. These are a participant identifier, a session identifier, number of continuous typing runs (i.e. 5), the duration of each run (i.e. 10 min), and the break duration between two runs (i.e. 2 min). The researcher simply has to provide this information and then hand over the device to the participant, and the participant simply has to press START to begin the experiment. From then onwards, the app will guide the participant through the experiment.

The app will read stimuli phrases from the provided phrase set; will provide a randomized copy of it to the participant. Randomization is performed using the Fisher-Yates shuffling algorithm (Fisher & Yates, 1948). The phrases come from the Enron Mobile Email Dataset (Keith Vertanen & Kristensson, 2011) which is described in Materials section.

As shown in the figure below, the participant will be shown an interface where he/she will be required to enter the phrase shown. The countdown timer will keep counting down from the specified run duration value to achieve this). Two timestamps will be captured and written to a file in the background, one corresponding to the first letter typed in the text box, and the second being the time pressed NEXT. Following this, the stimuli and response phrases will also be written to the file. When the user presses NEXT, the app will display the next sentence and clear the textbox. By this time, if the countdown timer has reached zero, pressing NEXT would take him to another activity which would allow the participant to take a break for the aforesaid time. This process is then repeated until the all the runs are complete.



ExperimentOne

University Username: test

Session Number: 10

MainActivity

Time Left 09:47

test - Session 10 - Part 1 of 5

banks want to exclude goodwill

banks want to exclude

MiddleActivity

Please take a break!

Time Remaining 01:54

Please fill the Break Questionnaire

Break No 1

Figure 13 - Software apparatus used for the study

Certain buttons (i.e. Back Key) are disabled by the app to avoid unexpected behaviour that would hinder the experiment. The app requires permission to vibrate and play ringtones, as the user needs to be notified when the break is over and it is time to start entering text again.

3.5 Materials

This section describes the surveys used, the phrase set used for the above study, compensation and phrase set.

3.5.1 Surveys

In addition to capturing the user's performance when using either type of keyboard, we also surveyed their responses on previous typing experience, mobile phone experience, smartphone experience, and perceived performance and user experience, which shall be explained in the upcoming sub sections.

3.5.1.1 Preliminary Survey

This was given at the beginning of the entire study, before the user had the opportunity to enter any text whatsoever. The purpose of the survey was to gauge the prior experience of the user.

Q1. In your life, which text entry method did you use more during your day-to-day activity?

Only Tapping	1	2	3	4	5	6	7	Only Gesturing

Rationale (Q1):

In conjunction with the final survey (presented at the end of the study), this will reveal if users had prior experience with gesture keyboard, if not, would have started using gesture keyboard in their day to day life as a result of their study.

Q2. What kind of mobile devices do you use? Tick all that apply.

Smartphone – Android
Smartphone – Apple
Smartphone – Microsoft
BlackBerry
Feature phone (no large touchscreen)
Tablet – Android
Tablet – Apple
Tablet – Microsoft
Phablet – Android (A very large smartphone)

Q3. Please write the brands and models of the mobile device you have used the most in last few years.

Brand	Model	Duration of Use

Q4. Please rate your ability to type using your mobile device.

Very slow typist	1	2	3	4	5	6	7	Very fast typist
Inaccurate typist	1	2	3	4	5	6	7	Very accurate typist

Rationale (Q2-Q4):

This is to gauge the user's previous smartphone experience, which is directly attributable to one's performance on a STK and SG. This will be revisited in the participants section.

3.5.1.2 Surveys during the studies

At each session, we gauged the participant's "in-situ" user experience via three surveys, one at the start of each session, one during the session breaks – with 4 sections, 1 for each break, and one at the end of each session.

3.5.1.2.1 Pre-survey

As the sessions were spread over multiple days, we wanted to find out if the initial results as provided in the preliminary survey had changed over time. Thus the following two questions were presented again:

Q5. Between now and the last experiment, which text entry method did you use more during your day-to-day activity?

Only Tapping 1 2 3 4 5 6 7 Only Gesturing

Q6. Please rate your current ability to type using your mobile device now (May have changed due to this experiment)

Very slow typist 1 2 3 4 5 6 7 Very fast typist

Inaccurate typist 1 2 3 4 5 6 7 Very accurate typist

Rationale (Q5-Q6):

If the responses to these questions differed from the previous ones, then that means the study has affected the participants in a positive (or negative) manner.

3.5.1.2.2 In-survey

This was provided during the break times provided. This was just to gauge the participant's perceived performance and their user experience.

Q7. How fast did you type during the session?

Very slow 1 2 3 4 5 6 7 Very fast

Q8. How many errors did you make during this session?

A lot of errors 1 2 3 4 5 6 7 No errors

Q9. How much do you like this typing method?

Not at all 1 2 3 4 5 6 7 Very much

Q10. How easy do you find this typing method?

Very hard 1 2 3 4 5 6 7 Very easy

3.5.1.2.3 Post-survey

This was given at the end of each session, again to gauge the participant's perceived observation how much they have either improved or worsened as a result from the study.

Q11. How much have you improved (or not) since last session (4 means = the same)?

Worsened 1 2 3 4 5 6 7 Improved

Q12. Did you find this session easier / more likeable than the last session (4 = the same)?

Much Harder 1 2 3 4 5 6 7 Much Easier

Q13. What posture did you use mostly for this session?

Thumb Single-Finger Two-Thumbs

Rationale (Q13):

As the hand posture does affect the study, as mentioned overleaf, we wanted to capture which hand postures were used by the participant for this particular session. They were told to be consistent when typing inside each session i.e. not to change their hand posture, and this was adhered to.

3.5.1.3 Final Survey – End of the study

This was the final questionnaire at the end of the full study. We required the participants to provide qualitative and open ended answers on what they liked and disliked about each input method.

Q14: What did you like about each input method?

Q15: What did you dislike about each input method?
Use your own words and be descriptive as possible. Talk about ease of use, learning curve, speed, accuracy and your user experience (did you feel fatigue after typing for one hour, did you keep getting faster / slower / more accurate / inaccurate etc.)

Q16: Did your everyday text input get affected as a result of this experiment?
(Did you learn a new method of text input (i.e. gesture), became aware about a new tool (Google keyboard), apply it to your own day-to-day life, or did you become faster, more accurate etc.)

Q17: What do you think about this experiment?

Rationale (Q16):

We wanted to find out if this experiment has affected the user's general typing experience in real life. As there were users who were exposed to SGK for the first time, it was possibly the most interesting question in this survey by far. We will revisit this question more in the results section.

3.5.2 Phrase Set

We used a subset of the Enron mobile email dataset (Keith Vertanen & Kristensson, 2011) with the following conditions:

- Each sentence should be less than 60 characters in length
- No numbers
- No special symbols

This resulted in phrase set of 1008 phrases. We counted 1,457 unique words in this test set. The rationale behind this was we didn't want users to switch between keyboards to enter numbers and special symbols – i.e. in Android, when using Google Keyboard; users have to change the view back and forth to enter numbers, symbols and letters. We decided this should be explored in a different study instead of this one.

3.5.2.1 Out of Vocabulary (OOV) words

We did not, however, exclude sentences with Out of Vocabulary words (OOVs). We did this because we want to find out how each keyboard type (STK, SGK) performed differently when OOV words were part of the mix. As SGK had no way of inferring OOV words, it was an interesting observation as to how it affected the typing speed, the error rate, the user experience and most of all how users dealt with this particular issue when typing.

We compared all the words in the phrase set against a standard lexicon (64K common words used in the English language). The words that weren't in the lexicon were each entered carefully on the Google keyboard, by tapping the centre of each key on the STK and by gesturing from the centre to centre of each key on the SGK. We noted that the same 44 words were out of vocabulary (OOV) words for both the STK and SGK. These OOVs appeared in 45 sentences (4.46% of 1,008) in the phrase set, and were marked as sentences with OOV words. These OOV sentences were analysed in post-hoc analyses after the experiment.

3.5.2.2 Non OOV Samples

The following are a few sample phrases from the set which do not include OOV words

> "I have received your messages and will respond accordingly"
>
> "Please make sure Bob Kelly is on the list"
>
> "I was answering Janet's comment"
>
> "Anything exciting going on today"
>
> "Email the consent to me"

3.5.2.3 OOV Samples

The following are a few sample phrases from the set which include OOV words

“Check with Vince Strohmeyer”

“If so Whitt is done”

“Why don't you ask Shanna Funkhouser for the details”

“It's death or dynegy with no clear leader”

“Are Linde and Kim available to assist rod”

3.5.2.4 Ordering of Phrases

The Fisher Yates Shuffling Algorithm (Fisher & Yates, 1948) was used to randomize the order of the phrases when presented to the participants. The algorithm is a simple, in place shuffling mechanism and can be outlined as follows using the Java programming language.

```
public static void fisherYatesShuffle(int[] ar) {
    // generate a randomizer
    Random rnd = ThreadLocalRandom.current();

    for (int i = ar.length - 1; i > 0; i--) {
      // generate random index between 0 and (i+1)
      int index = rnd.nextInt(i + 1);

      // perform a simple swap between the current and
      // random positions
      int a = ar[index];
      ar[index] = ar[i];
      ar[i] = a;
    }
}
```

3.6 Compensation

Participants were compensated £50 for their time in amazon vouchers. The standard rate of compensation in University of St Andrews is £5 per hour for participating in experiments. Given that each participant had to attend 10 sessions of 1 hour each, their commitment was 10 hours.

Further we offered an incentive of an extra £15 for the fastest typing participant in each keyboard type, under a certain error rate threshold.

3.7 Participants

We recruited 12 volunteers from the University of St Andrews campus, the details of whom are described in the sections below. The rationale for 12 participants being sufficient was based on previous longitudinal studies conducted with a similar statistical analyses (**α=0.05)**, peer reviewed by the HCI community and published (P. O. Kristensson & Denby, 2009).

3.7.1 Participant Demographics

Due to the ethics agreement we cannot publish any identifiable information about the participants - therefore the aggregate results of each demographic will be described below and not attributed to individual participants.

3.7.1.1 Gender

We had an equal number of males and females – 6 participants each.

3.7.1.2 Age

The age range was between 21-34, with the mean age being 25 and Standard Deviation being 4. This ensured we had a satisfactory distribution of ages which is quite representative of the real world population who uses smartphones for text entry in the year 2013 (when this experiment was performed).

3.7.1.3 English Proficiency

Four participants were native English speakers, and the rest used English as their second language. Given they were all doing either a undergraduate, postgraduate or PhD in University of St Andrews they had to be proficient in English, if not they would not be admitted for study – as per the English language requirements of university admissions - getting a 7.0 or above in IELTS (“IELTS,” n.d.). This ensured that our participants were able to understand, read and copy the sentences in the above phrase set without difficulty.

3.7.1.4 Geographic Distribution

The best part about running studies in University of St Andrews is that it attracts students from all over the world. In a recent survey, it was found that St Andrews students represent 120 different cultures, which gives a mini sample of the global population. Our study therefore, had participants from 10 different countries.

Greece	Germany	USA	Sri Lanka	Pakistan	India	Scotland	Nigeria	Bulgaria	England
1	1	2	1	1	1	1	2	1	1

3.7.1.5 Field of Study Distribution

The participants field of study also varied across the disciplines – this also ensures that our study was rich in terms of different levels of technical expertise and not include participants from either a daily high tech usage demographic (e.g. Computer Science) or low (e.g. the Humanities). The participants' fields of study can be summarised as follows.

Computer Science	Other Sciences	Arts & Humanities
6	1	5

Further, the participants' level of study can be summarised as follows.

Undergraduate 1st Year	Undergraduate 2nd Year	Undergraduate 3rd Year	Undergraduate 4th Year	Masters	PhD
2	2	1	1	3	3

3.7.1.6 Smartphone Experience

By interviewing the participants, we gauged their previous smartphone experience and found the following. This also ensures we had a satisfactory variation/distribution among participants with regards to previous smartphone experience.

Android Smartphone or Tablet	iPhone iPod	Nokia Lumia Windows Phone	BlackBerry	No Smartphone Experience (but with T9)
5	3	1	2	2

3.7.1.7 Exposure to STK & SGK

Again by interviewing the participants, we gauged their previous experience with STK and SGK, which is probably the most relevant demographic survey related to this study.

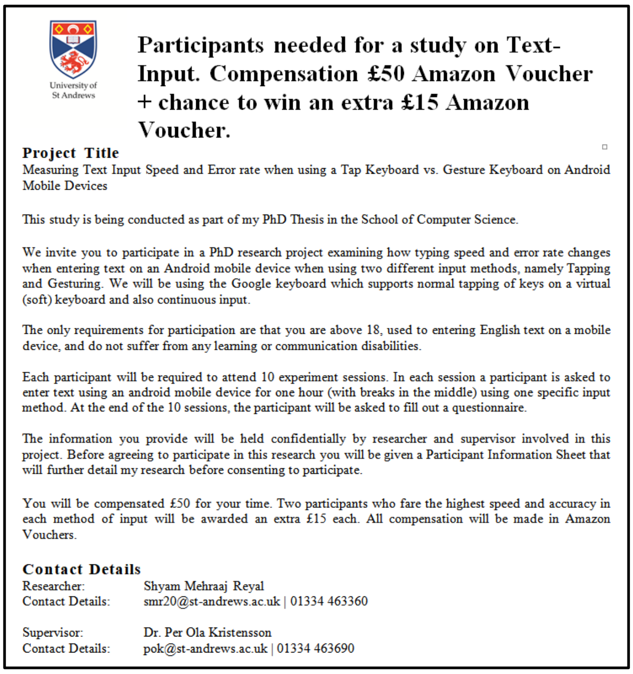
Experience with QWERTY	Experience with STK	Experience with SGK
all	10	3

3.7.2 Design

We incorporated a within-subjects design for this study - meaning each participant had to experience both conditions (STK and SGK). To minimise starting bias, we had 6 participants use STK first, and the remaining 6 participants to use SGK first. This ensured that we had a balanced group of participants with fully balanced conditions between the two groups. The participants were allocated randomly to the two groups.

3.7.3 Recruitment

We used various channels to advertise this study and recruit participants, briefly outlined below. A sample advert used for this study is shown in the figure.



University of St Andrews

Participants needed for a study on Text-Input. Compensation £50 Amazon Voucher + chance to win an extra £15 Amazon Voucher.

Project Title
Measuring Text Input Speed and Error rate when using a Tap Keyboard vs. Gesture Keyboard on Android Mobile Devices

This study is being conducted as part of my PhD Thesis in the School of Computer Science.

We invite you to participate in a PhD research project examining how typing speed and error rate changes when entering text on an Android mobile device when using two different input methods, namely Tapping and Gesturing. We will be using the Google keyboard which supports normal tapping of keys on a virtual (soft) keyboard and also continuous input.

The only requirements for participation are that you are above 18, used to entering English text on a mobile device, and do not suffer from any learning or communication disabilities.

Each participant will be required to attend 10 experiment sessions. In each session a participant is asked to enter text using an android mobile device for one hour (with breaks in the middle) using one specific input method. At the end of the 10 sessions, the participant will be asked to fill out a questionnaire.

The information you provide will be held confidentially by researcher and supervisor involved in this project. Before agreeing to participate in this research you will be given a Participant Information Sheet that will further detail my research before consenting to participate.

You will be compensated £50 for your time. Two participants who fare the highest speed and accuracy in each method of input will be awarded an extra £15 each. All compensation will be made in Amazon Vouchers.

Contact Details

Researcher:	Shyam Mehraaj Reyal
Contact Details:	smr20@st-andrews.ac.uk | 01334 463360
Supervisor:	Dr. Per Ola Kristensson
Contact Details:	pok@st-andrews.ac.uk | 01334 463690

3.7.3.1 Screening

Participants were screened for:

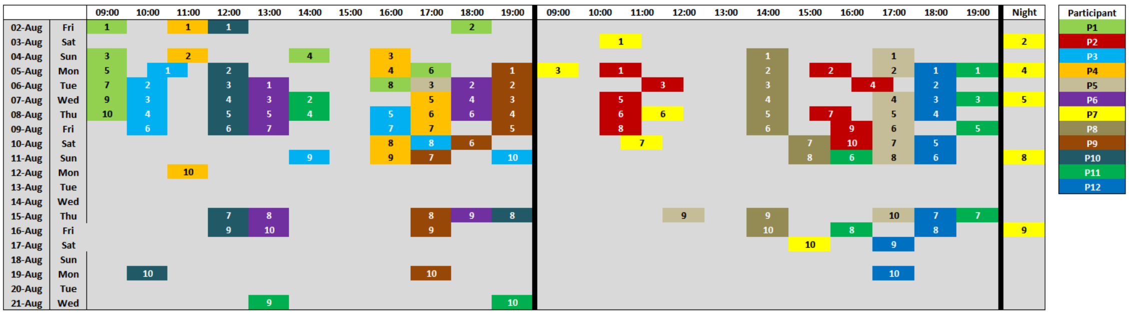
(a) English proficiency – either they had to be native English speakers or use English as a second language in their day to day life / studies

(b) Experience with QWERTY – since we use this keyboard layout, participants had to be experienced in typing using a QWERTY keyboard either using the computer or their mobile device

(c) Experience with mobile phones – participants had to have some mobile device experience (even a pre-smartphone era mobile device with a T9 keypad), as we decided that participants with zero mobile phone experience would significantly skew the results of this study.

3.7.4 Scheduling

The scheduling of participants was done as shown in the figure below. The participants had to attend for 10 sessions in total – 5 sessions in each condition (STK & SGK). Each session was spaced at least 4 hours apart and at most 2 days. Since the participants were mostly subject to an academic calendar, they preferred coming on the same time slot of the day e.g. 9.00am-10.00am in the case of Participant 1. Also we had two mobile devices so two sessions could be scheduled in parallel.



		09:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	09:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	Night
02-Aug	Fri	1		1	1						2													
03-Aug	Sat													1										2
04-Aug	Sun	3		2			4		3									1			1			
05-Aug	Mon	5	1		2				4	6		1	3	1				2	2		2	1	1	4
06-Aug	Tue	7	2		3	1			8	3	2	2			3			3		4		2		
07-Aug	Wed	9	3		4	3	2			5	4	3		5				4			4	3	3	5
08-Aug	Thu	10	4		5	5	4		5	6	6	4		6	6			5	7		5	4		
09-Aug	Fri		6		6	7			7	7		5		8				6		9	6		5	
10-Aug	Sat								8	8	6				7				7	10	7	5		
11-Aug	Sun						9		9	7		10							8	6	8	6		8
12-Aug	Mon			10																				
13-Aug	Tue																							
14-Aug	Wed																							
15-Aug	Thu				7	8				8	9	8				9		9			10	7	7	
16-Aug	Fri				9	10				9								10		8		8		9
17-Aug	Sat																		10		9			
18-Aug	Sun																							
19-Aug	Mon		10							10											10			
20-Aug	Tue																							
21-Aug	Wed					9						10												

Participant
P1
P2
P3
P4
P5
P6
P7
P8
P9
P10
P11
P12

3.8 Procedure and Execution

The execution of the study was done in two steps. The first was a pilot where the authors of the paper used the apparatus to find any errors in the implementation that could affect the study, followed by the actual study involving the 12 participants.

3.8.1 Pilot Study

The study apparatus was piloted intensely before the actual study commenced, and a number of problems were identified – which were (fortunately) not problems with the study design but the android implementation. We were able to fix these with ease before the actual study commenced. The most significant problem which resulted in a study design change is highlighted below.

3.8.1.1 Timer and Phone Orientation

The timer on the software apparatus reset every time the user switched between portrait and landscape mode. This was due to the android platform killing and reloading the Activity every time the phone orientation changes. This could be overcome in two ways

i. Instead of saving the timer state to a variable, it could be stored and loaded from SharedPreferences which is a form of persistent storage on Android

ii. Lock the phone to portrait mode – we decided to go with this one as this would prevent another confounding variable being introduced into the study.

3.8.2 Actual Study

As explained above, the experiment consisted of ten sessions split into five sessions for STK and five sessions for SGK. We divided the participants into two equal groups. Participants in the first group completed their first five sessions using the STK and the last five sessions using the SGK. The other group had the opposite order. Before commencing the first and the sixth sessions, participants were given time to familiarize themselves with the new text entry method if they hadn’t used it in the past. The sessions were spaced at least four hours apart and were maximally separated by two days. Each session consisted of five 10-minute-long typing runs followed by two-minute-long breaks.

The experiment used a transcription task where participants were shown a phrase from the dataset and asked to copy it. We encouraged participants to focus on both speed and accuracy by providing an additional £15 Amazon voucher as an incentive to the fastest and the most accurate participants. Whilst being encouraged to use the Google keyboard’s suggested words for correction, we discouraged participants to go back and correct errors unless absolutely necessary. I.e. when transcribing the phrase, if an error or typo was made, we asked the participants to decide if they would send this message off if a human user was listening on the other end. If they decided the original word was still able to be deciphered by the recipient, then they were not required to go back and correct it. Participants were seated during the experiment, with no distractions from the environment. Our experiment app recorded the stimulus phrases and the response text using millisecond timestamps when the user entered the first character and when the user pressed NEXT.

Participants rated their previous experience with software keyboards (STK and SGK) on mobile devices, and self-rated themselves on how fast and accurate they thought they were. During each two-minute break, they were asked to rate the speed, accuracy, preference, and ease of use of the currently used text entry method. Answers were recorded on a 1–7 Likert scale.

We intentionally did not control hand posture. Instead we asked participants to use their preferred posture and report it at the end of each session. The choices were single thumb, single finger and two thumbs. At the end, participants were asked to write descriptive and open comments about what they liked and/or disliked about each text entry method.

3.9 Results & Analysis

Using STK, participants entered an average of 1393 sentences (SD = 275) during each session totalling 13,927 data points. 211 of these were filtered out as outliers (being more than 3 standard deviations away from the mean). Using SGK, participants entered an average of 1,282 sentences per session (SD = 225), which totalled 12,816 data points; out of which 278 points were discarded as outliers (using same principle as above).

In total we collected 100 hours of data – 50 minutes of typing (excluding breaks) x 120 sessions. These are the breakdowns of the data points. Outliers were filtered as being more than three standard deviations from the mean.

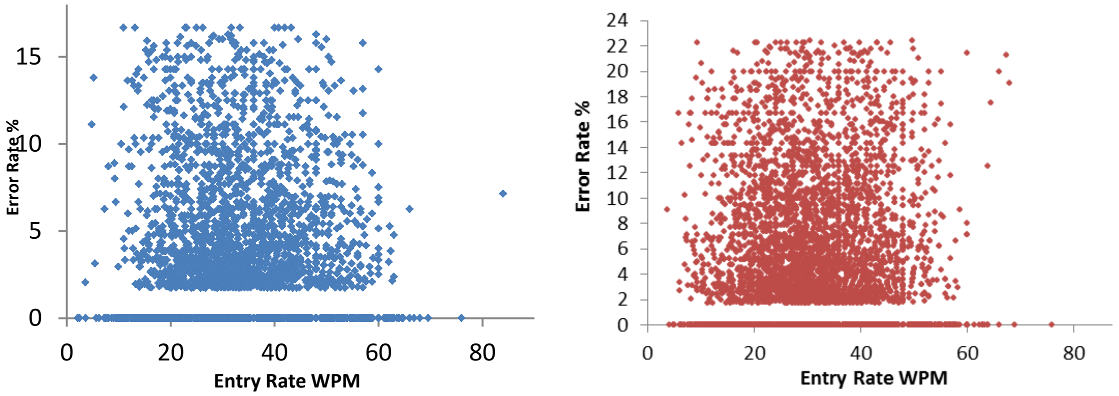


Figure 14 - Entry Rate vs Error Rate Scatter Plots for STK and SGK

	Data Points	Outliers	Data Points per User
STK	13,927	211	1393 (SD=275)
SGK	12,816	225	1282 (SD=225)

3.9.1 Entry Rate

As mentioned in the Variables section above, entry rate was calculated as words-per-minute; with a word defined as five consecutive characters (spaces were also considered as characters). The number of characters entered was derived from the length of the response String. The time required to enter each phrase was defined as the interval between when the participants entered the first character and pressing the NEXT button on the interface, and was calculated in seconds.

As expected, participants became faster with practice. (See figure 13). In the first session the mean entry rate for STK was 29.1 WPM (SD~6.4) and for the last session it was 32.8 WPM (SD~9.1). For SGK the mean entry rate in the first session was 25.4 WPM (SD~5.5) and 30.6 WPM (SD~6.0) in the last session. The mean entry rate for SGK increased at a faster rate with practice than for STK – i.e. at the beginning STK was observed to be significantly faster than SGK, but at the end it became not significant.

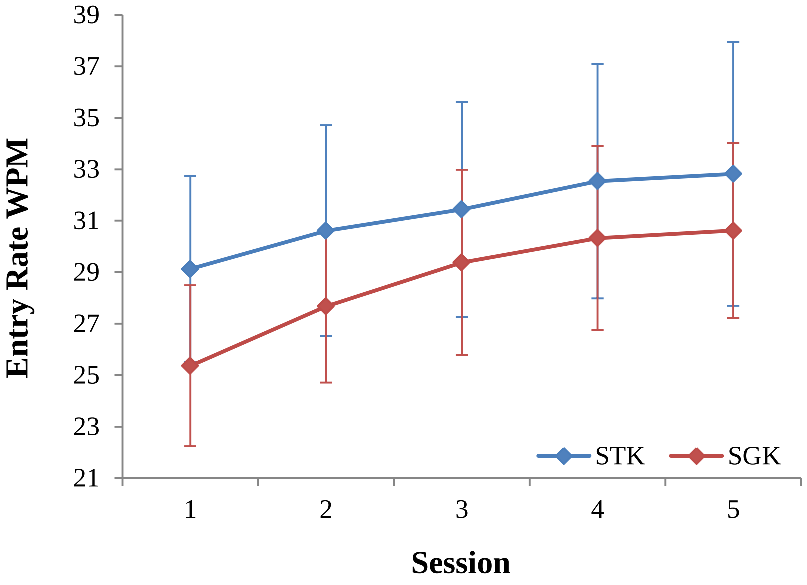


Figure 15 - Entry Rate vs Session for STK and SGK in Experiment A

WPM	*m* S1	95% CI S1	*m* S5	95% CI S5
STK	29.1	25.5 – 32.7	32.8	27.7 – 37.9
SGK	25.4	22.2 – 28.5	30.6	27.2 – 34.0
ANOVA	F_{df}	df	η_p^2	p
Input Type	5.406	1,11	.330	**.040***
Session	22.036	4,44	.667	**.000***
Input × Session	0.818	4,44	.069	.521

3.9.2 Character Error Rate

SGK resulted in a significantly higher error rate (2.04–2.34% CER) than STK (1.09–1.11% CER) as shown in the table. The error rates did not change over time and the interaction between input method and session was also not significant.

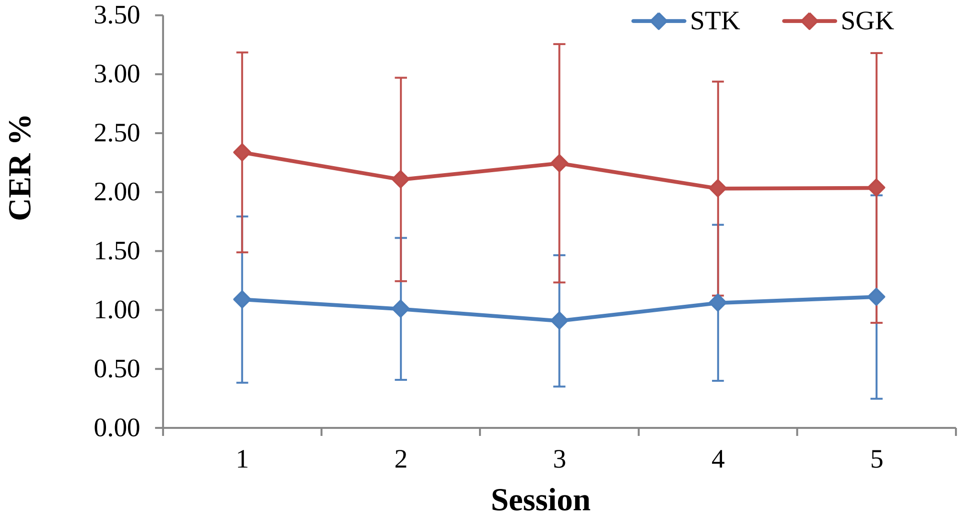


Figure 16 - Character Error Rate vs Session for STK and SGK in Experiment A

CER	*m* S1	95% CI S1	*m* S5	95% CI S5
STK	1.09	0.38 – 1.79	1.11	0.25 – 1.97
SGK	2.34	1.49 – 3.18	2.04	0.89 – 3.18
ANOVA	F_{df}	df	η_p^2	p
Input Type	17.267	1,11	.611	**.002***
Session	0.397	4,44	.035	.810
Input × Session	0.669	4,44	.057	.617

3.9.3 Word Error Rate

The word error rate followed a very similar pattern to the CER, but with higher values.

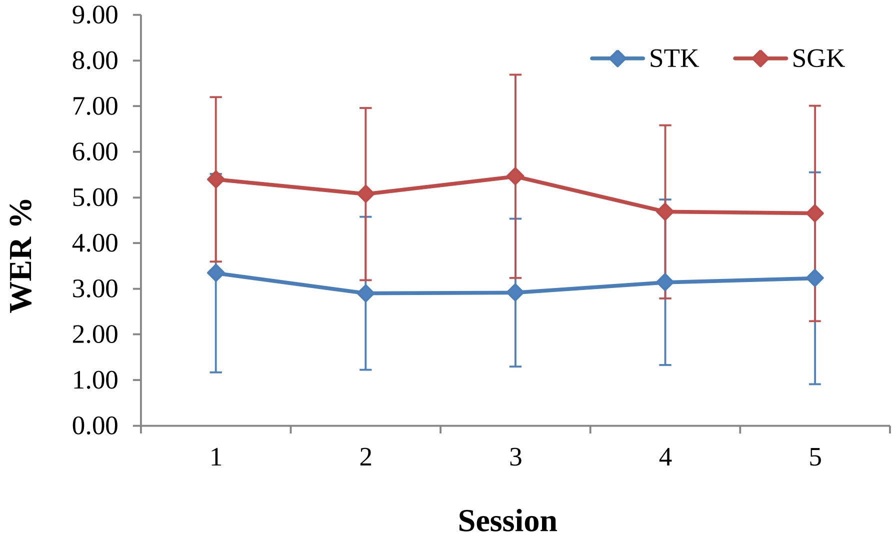


Figure 17 - Word Error Rate vs Session for STK and SGK in Experiment A

3.9.4 Excluding Sentences with OOV Words

We identified 1,131 data points containing OOV sentences (4.31% of 26,254). Recall that all the OOVs in the study affect both STK and SGK.

3.9.4.1 Entry Rate

Excluding OOV sentences results in a narrowing of the entry rate of STK and SGK and the difference is no longer significant.

WPM	*m* S1	95% CI S1	*m* S5	95% CI S5
STK	29.4	25.7 – 33	33.1	27.9 – 38.3
SGK	25.8	22.7 – 28.9	31.2	27.8 – 34.6
ANOVA	F_{df}	df	η_p^2	p
Input Type	3.946	1,11	.264	.072
Session	22.376	4,44	.670	**.000***
Input × Session	1.071	4,44	.089	.382

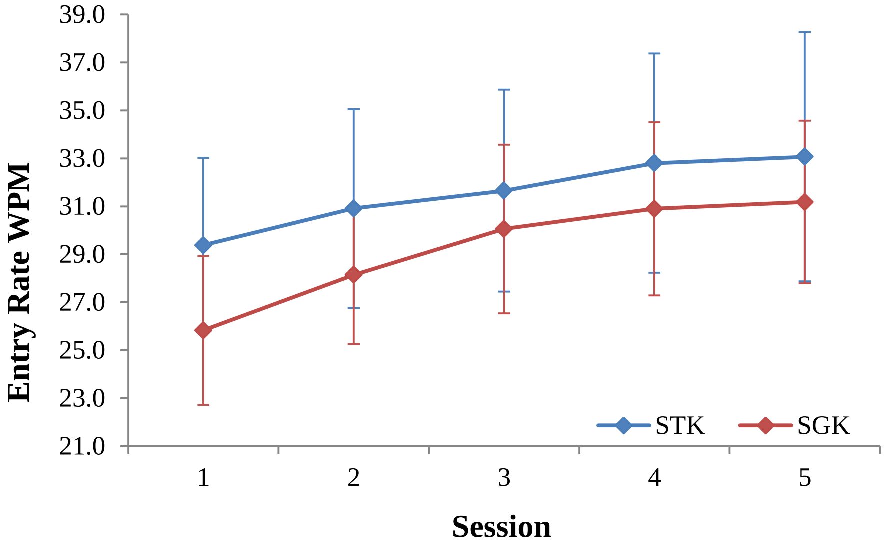


Figure 18 - Entry Rate vs Session for STK and SGK in Experiment A – excluding OOV words

3.9.4.2 Character Error Rate

CER also dropped slightly but the difference between STK and SGK is less marked.

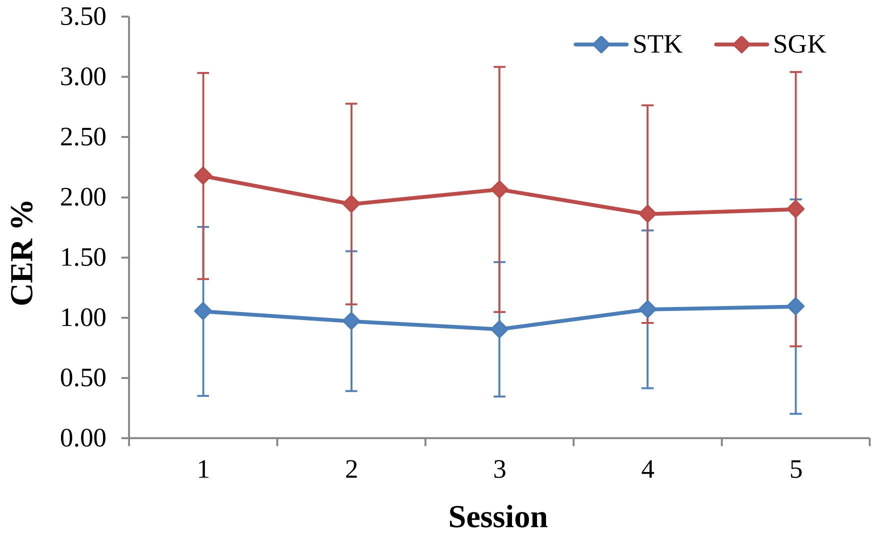


Figure 19 - CER vs Session for STK and SGK in Experiment A – excluding OOV words

CER	*m* S1	95% CI S1	*m* S5	95% CI S5
STK	1.05	0.35 – 1.76	1.09	0.2 – 1.98
SGK	2.18	1.32 – 3.03	1.90	0.76 – 3.04
ANOVA	F_{df}	df	η_p^2	p
Input Type	12.429	1,11	.530	**.030***
Session	0.343	4,44	.030	.848
Input × Session	0.572	4,44	.049	.685

3.9.4.3 Word Error Rate

The WER followed a similar pattern to the CER, but with higher values.

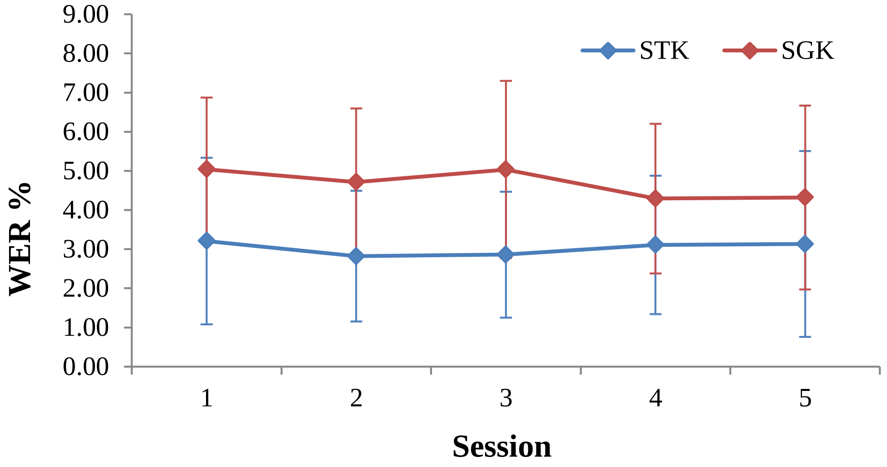


Figure 20 - WER vs Session for STK and SGK in Experiment A – excluding OOV words

3.9.5 Only Investigating Sentences with OOV Words

We investigated the previously excluded 1,131 data points, which consisted only of OOV sentences.

3.9.5.1 Entry Rate

The STK was significantly faster than SGK when participants entered sentences with OOVs. While both conditions have been affected by OOVs, SGK was penalized more.

WPM	*m* S1	95% CI S1	*m* S5	95% CI S5
STK	25.2	21.7 – 28.7	28.9	24.6 – 33.2
SGK	18.8	15.5 – 22.2	22.6	18.6 – 26.7
ANOVA	F_{df}	df	η_p^2	p
Input Type	35.929	1,11	.766	**.000***
Session	6.132	4,44	.358	**.001***
Input × Session	0.303	4,44	.027	.874

A clear significant difference in entry rate was observed in the statistical analysis, followed by participants clearly being improving their entry rates over the sessions. The interaction between the input type x session was not significant.

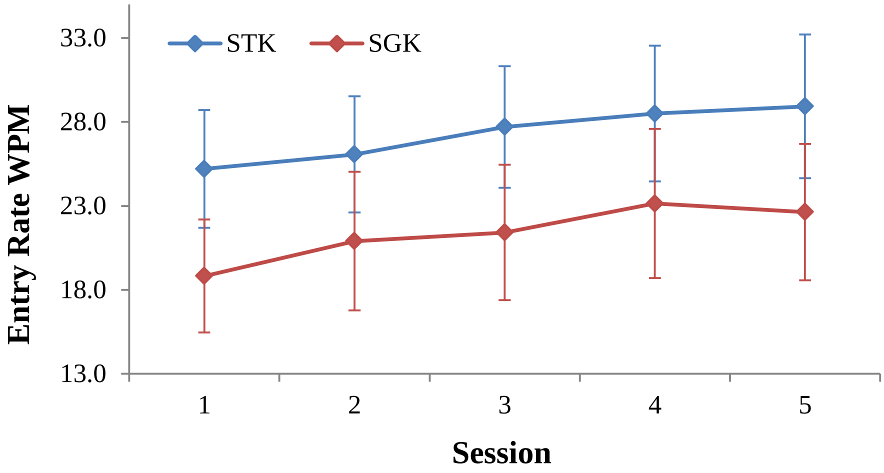


Figure 21 - Entry Rate vs Session for STK and SGK in Experiment A – considering only OOV sentences

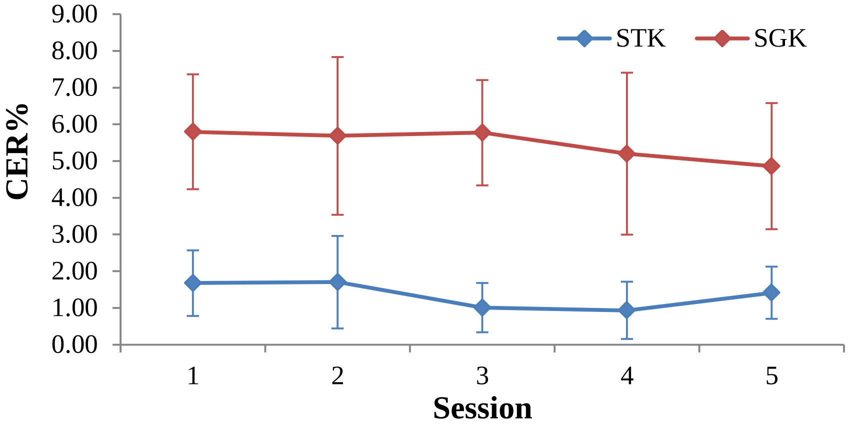


Figure 22 – CER vs Session for STK and SGK in Experiment A – considering only OOV sentences

CER	*m* S1	95% CI S1	*m* S5	95% CI S5
STK	1.68	0.78 – 2.57	1.41	0.7 – 2.13
SGK	5.80	4.23 – 7.36	4.86	3.14 – 6.57
ANOVA	F_{df}	df	η_p^2	p
Input Type	48.2	1,11	.814	**.000***
Session	0.955	4,44	.080	.441
Input × Session	1.055	4,44	.087	.390

There were very high error rates are reported in both conditions, but they are significantly higher in SGK. Overall OOVs present more challenges to SGK than STK.

3.9.5.2 Word Error Rate

WER also followed a similar pattern to CER but with higher values.

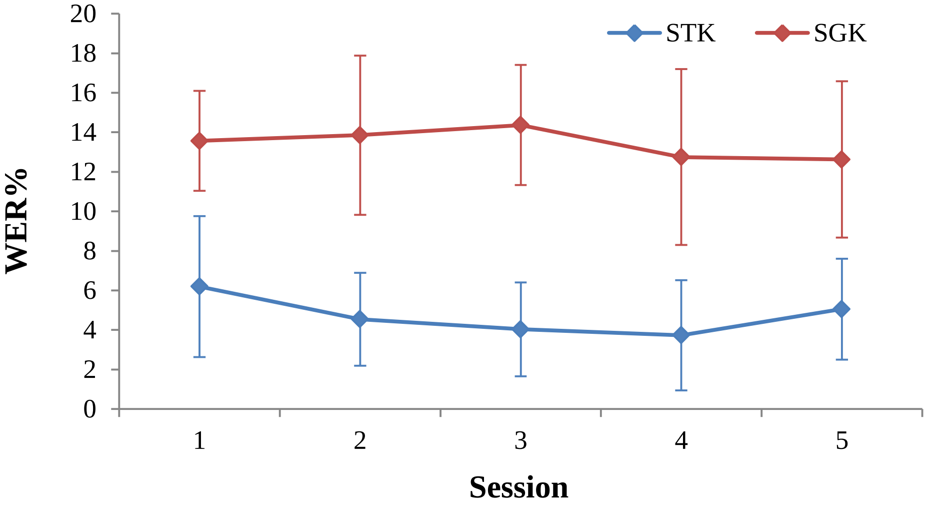


Figure 23 - WER vs Session for STK and SGK in Experiment A – considering only OOV sentences

3.9.6 Analysis of Hand Postures

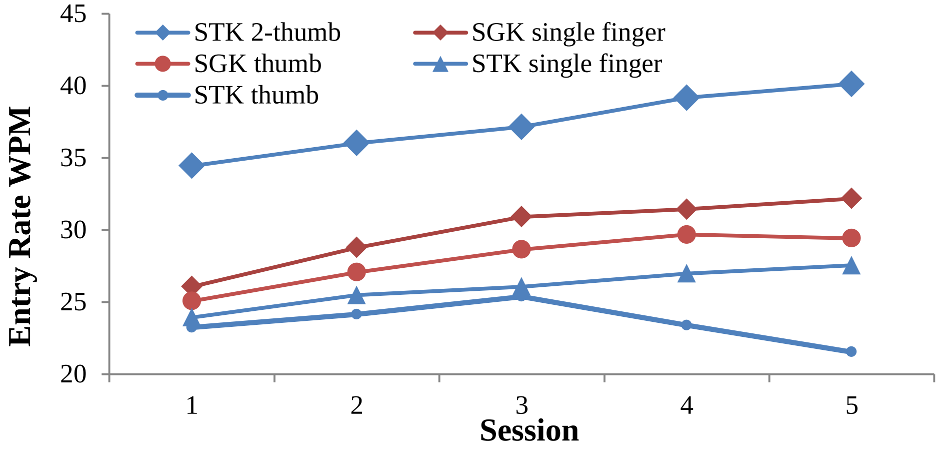


Figure 24 - Entry vs Session for Hand Postures in Experiment A

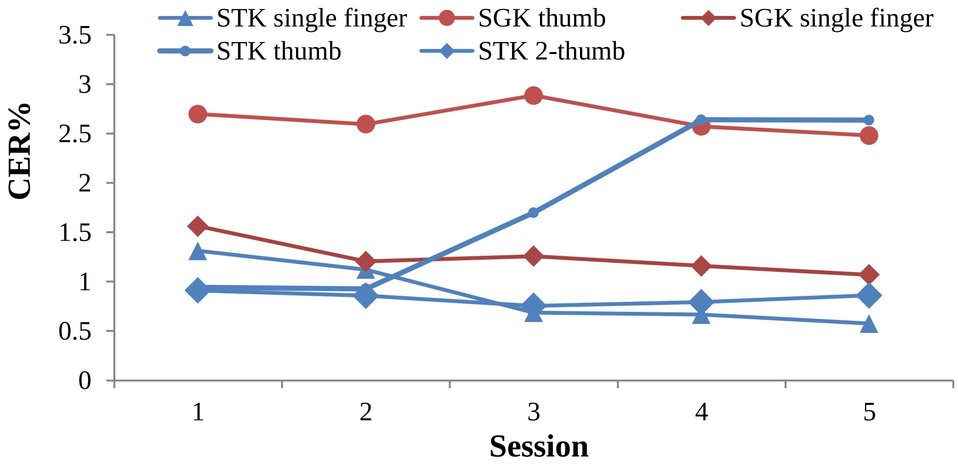


Figure 25 - CER vs Session for Hand Postures in Experiment A

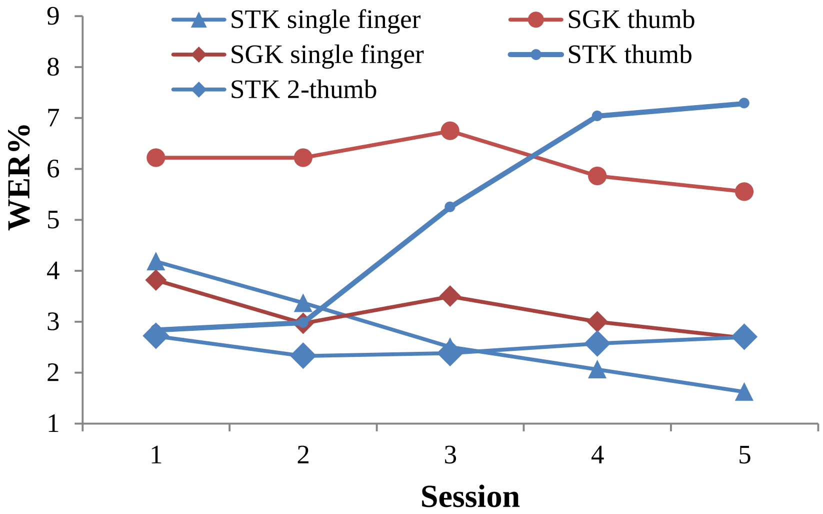


Figure 26 - WER vs Session for Hand Postures in Experiment A

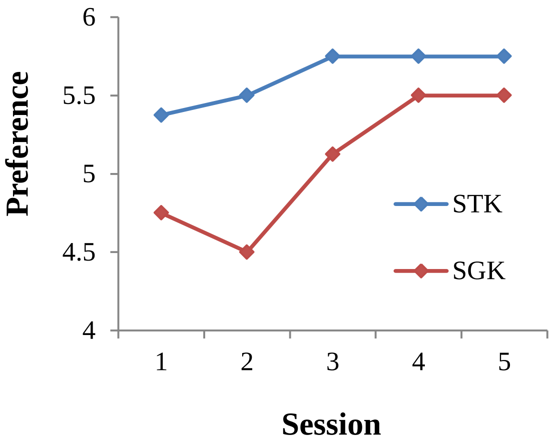
This indicates that two-thumb STK was the fastest, closely followed by single finger SGK. The difference in error rate between the different hand postures within STK was complex. Within SGK, single finger SGK produced lower error rates than single-thumb SGK. These results are indicative only, as a) hand postures were not controlled in the experiment, b) hand postures were self-reported by the participants, and c) some participants varied their hand postures across sessions.

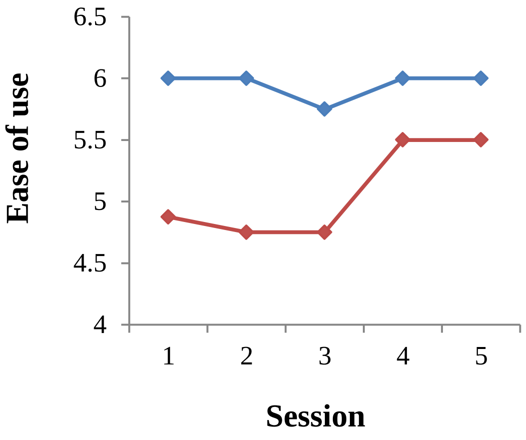
For these reasons, we do not report results of statistical analyses for hand postures. However, the data suggest hand posture might be an important factor for complete understanding of STK and SGK performance. Moreover, our data indicates that hand postures might have different effects on STK and SGK.

3.9.7 User Ratings

We calculated the median Likert-scale ratings from the participants' subjective ratings provided during their two-minute breaks between typing runs. Friedman's test revealed that across sessions participants felt their text entry experience became faster, more accurate, easier and more preferable with SGK over the sessions.

This was not the case with STK, whose plots were more flat. Users rated STK as significantly faster, easier to use and more preferred over SGK. However, they didn't find it significantly more accurate.





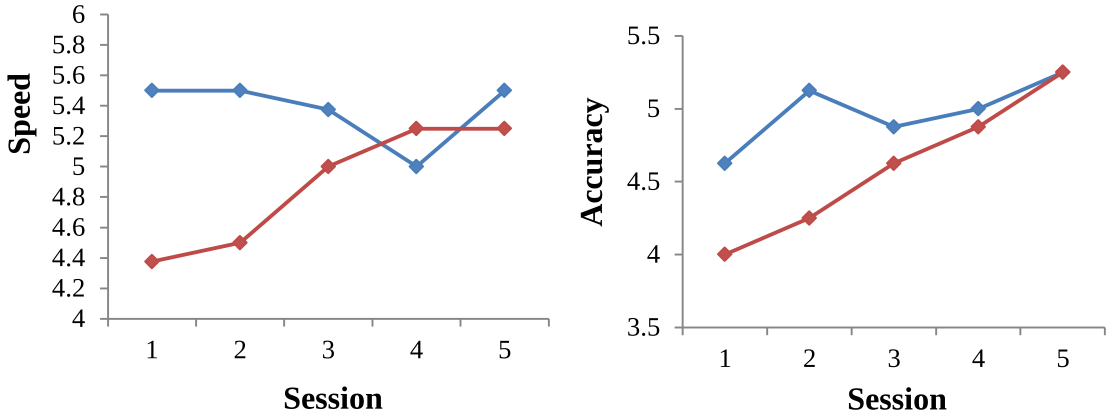


Figure 27 –User Ratings vs Session for Perceived Performance and User Experience in Experiment A

User Rating	STK		SGK	
	χ2(4)	*p*	χ2(4)	*p*
Input Speed	2.069	.723	15.584	**.004***
Accuracy	7.948	.093	13.083	**.011***
Ease of use	6.450	.168	14.530	**.007***
Preference	6.996	.136	14.231	**.006***

User Rating	Median		Friedman's Test Statistics	
	STK	SGK	χ2(1)	*p*
Input Speed	5.5	5	6.231	**.013***
Accuracy	5	4.75	2.200	.138
Ease of use	6	5	5.453	**.020***
Preference	5.5	5	7.681	**.006***

3.9.8 Open Comments

Participants also contributed open comments on what they liked and disliked about each input method. Representative comments for and against each text entry method are as shown below. A few participants added general comments such as: "Speed and accuracy depends upon how tired you are and your mental state" and "Started to use gesture input on a daily basis".

3.9.8.1 Like about STK

1. “easier to input names, slang, abbreviations, and possible to change manually”
2. “faster, and gives more control over what is been typed”
3. “easier to correct errors“
4. “fast and very easy to learn. Gives freedom of using two thumbs to type and select corrections and predictions easily”
5. “more convenient as can use two thumbs or fingers to type”
6. “much less fatigue than gesture keyboard”
7. “can put the phone on the table and type like a regular keyboard “
8. “feels more secure especially when walking”

3.9.8.2 Like about SGK

9. “new experience for me, really liked it “
10. “enjoyable for me to slide my finger instead of tapping “
11. “less exhausting than tapping “
12. “gives spaces between words automatically “
13. “fast in completing well known sentences and longer words “
14. “good with longer words, especially with repeated characters “
15. “nice for short words “
16. “requires less movement of fingers, a smooth curve instead of several tapings “
17. “words quickly become committed to muscle memory “

3.9.8.3 Dislike about STK

18. “more stressful “
19. “time taken to type a long word is annoying “
20. “accidently pressed space a lot of times instead of bottom row keys “
21. “wrong buttons are pressed very easily “
22. “I sometimes press the dot instead of the space “
23. “backspace button hit when trying to press L “

3.9.8.4 Dislike about SGK

24. “did not like the automatic spacing especially when there is a hyphen required “
25. “short words are repeatedly mistaken “
26. “problem when words contain two consecutive similar letters “

27. “difficult and time consuming to correct simple errors “
28. “very slow, accuracy decreases after 5 minutes of typing, very hard to learn, very exhausting and boring “
29. “difficult to input words with letters that are adjacent on the keyboard “
30. “by any chance if a single motion is incorrect, it will never guess the correct word “
31. “have to use more finger pressure especially when drawing long gestures “
32. “impossible to input a word not in the dictionary “
33. “fingers get tired quickly, and if your hands are wet, gets even more difficult “
34. “fatigue when typing for long hours “
35. “can’t add to previously typed words “
36. “phone size can be too big to do gesturing with single thumb “
37. “issues with proper nouns and unusual spellings “

3.9.8.5 General Comments

38. “speed and accuracy depends upon how tired you are and your mental state “
39. “Started to use gesture input on a daily basis “
40. “I am driven towards using gesture input as the sessions progress “
41. “I got faster in tapping over the sessions”

3.9.8.6 Analysis of comments

These comments can be broadly categorised into comments regarding error correction, and about general user experience. It is evident that most comments about disliking SGK are to do with error correction and OOV words, and liking SGK is to do with general user experience – see highlighted cells in the table below.

	Re: OOV and Error Correction	Re: General User Experience
Like about STK	1, 2, 3	4, 5, 6, 7, 8
Like about SGK		9, 10, 11, 12, 13, 14, 15, 16, 17
Dislike about STK	20, 21, 22, 23	18, 19,
Dislike about SGK	24, 25, 26, 27, 29, 30, 32, 35, 37	28, 31, 33, 34, 36

3.10 Summary

3.10.1 Entry Rates

In this table, we report the entry rates from the various analyses performed.

Mean Entry Rates	Normal	Excluding OOV Words	Only Sentences with OOV Words
STK	34.66 WPM (SD=10.2)	34.89 WPM (SD=10.2)	29.55 WPM (SD=8.80)
SGK	32.02 WPM (SD=9.64)	32.37 WPM (SD=9.50)	23.85 WPM (SD=9.33)
Is difference significant?	Yes	No	Yes
Improvement over the sessions?	Yes	Yes	Yes

So this tells us that, in a lab setting, under a normal phrase set mixed between OOV and non-OOV words, STK outperforms SGK in terms of entry rate. Further, participant's entry rate will improve with time. Also, we can see that the presence of OOV words impacts SGK more – i.e. when OOV words are removed, the difference in entry rate between STK and SGK becomes non-significant.

Therefore we have enough evidence to reject the following null hypotheses:

H0,a There is no difference in text entry rate between the two keyboards (STK and SGK) in a lab setting

H0,e The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of entry rate, in a lab setting

And replace them with the following alternate hypotheses:

H1,a STK has a faster entry rate than SGK in a lab setting

H1,e The use of Out of Vocabulary (OOV) words affect SGK more than STK differently in terms of entry rate, in a lab setting

3.10.2 Error Rates

In this table, we will consider only Character Error Rates (CER) as this was the basis for the statistical analysis. Word Error Rates were observed to follow a similar pattern with higher values.

Mean Error Rates	Normal	Excluding OOV Words	Only Sentences with OOV Words
STK	0.91% (SD=2.57)	0.89% (SD=2.55)	1.29% (SD=2.86)
SGK	2.05% (SD=4.27)	1.86% (SD=4.14)	5.49% (SD=5.60)
Is difference significant?	Yes	Yes	Yes
Improvement over the sessions?	No	No	Yes

So this tells us that, in a lab setting, under a normal phrase set mixed between OOV and non-OOV words, STK outperforms SGK in terms of error rate. Further, participants' error rate did not change with time for sentences without OOV words. When using sentences with OOV words, the participant's error rate actually increased with time.

Therefore we have enough evidence to reject the following null hypothesis:

H0,b There is no difference in character error rate after correction between the two keyboards (STK and SGK) in a lab setting

And replace them with the following alternate hypothesis:

H1,b SGK produces more errors than STK in a lab setting

However, we do not have enough evidence to reject the following null hypothesis, as with and without OOV words, there is still a significant difference in error rates between the two input methods.

H0,f The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of error rate, in a lab setting

3.10.3 Hand Postures

In this table, we summarise the results from the hand posture analysis.

Mean Entry Rates	Two Thumb	Single Finger	Single Thumb
STK	35.75 (SD=9.99)	25.69 (SD=7.73)	26.63 (SD=7.68)
SGK	-	34.75 (SD=9.85)	30.11 (SD=9.01)

Mean Error Rates	Two Thumb	Single Finger	Single Thumb
STK	0.75% (SD=2.82)	3.96% (SD=4.87)	0.79% (SD=2.33)
SGK	-	3.06% (SD=5.15)	1.03% (SD=3.34)

These results are indicative only, as a) hand postures were not controlled in the experiment, b) hand postures were self-reported by the participants, and c) some participants varied their hand postures across sessions. For these reasons, we do not report results of statistical analyses for hand postures.

Although, the data suggest hand posture might be an important factor for complete understanding of STK and SGK performance and we could conjecture the following (as indicative only):

- Hand postures might have different effects on STK and SGK.
- Two-thumb STK could be the fastest mechanism to type in a lab setting, with the smallest error rates, and probably the most popular
- The two-thumb SGK (or Bi-Manual SGK) is probably not a popular choice among users,
- In both methods, single finger produces more errors than single thumb
- In SGK, single finger performs faster than single thumb

However, given the non-controlled nature of this variable, the self-reporting, and not having statistical justification, this does not give us enough evidence to reject the null hypotheses:

H0,c The entry rate does not differ between different hand postures used, in a lab setting

H0,d The error rate does not differ between the different hand postures used (see H0,c) in a lab setting

3.10.4 Subjective Ratings

In this table, we summarise the results from the subject ratings and open comments

	Significant improvement over the sessions	Significantly better than the other
Input Speed	Yes	Yes – STK
Accuracy	Yes	No
Ease of use	Yes	Yes – STK
Preference	Yes	Yes – STK

From this table we can see that users saw an improvement in their perceived performance over the sessions, and the sessions became easier with time. Further, they rated STK higher than SGK in general in the lab sessions in terms of input speed, ease of use, and preference. When analysing the open comments, it is evident that users found STK to be easier when entering text with OOV words, and correcting errors.

Therefore we have enough evidence to reject the following null hypothesis:

H0,g The user experience of the participants did not differ between STK and SGK in a lab setting

And replace it with this alternate hypothesis:

H1,g Participants preferred STK over SGK in a lab setting

3.11 Conclusions

All in all, we could draw the following conclusions from this study:

H1,a STK has a faster entry rate than SGK in a lab setting

H1,b SGK produces more errors than STK in a lab setting

H0,c The entry rate does not differ between different hand postures used, in a lab setting
-- *not enough evidence to say otherwise*

H0,d The error rate does not differ between the different hand postures used (see H0,c) in a lab setting
-- *not enough evidence to say otherwise*

H1,e The use of Out of Vocabulary (OOV) words affect SGK more than STK differently in terms of entry rate, in a lab setting

H0,f The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of error rate, in a lab setting -- not enough evidence to say otherwise

H1,g Participants preferred STK over SGK in a lab setting

4

Study B – Comparison of STK and SGK in the Wild

4.1 Motivation

Even though STKs and SGKs have become the mainstream touchscreen text entry methods, the HCI research literature offers little empirical evaluation of the current state of affairs in general, and the performance and experience difference between STKs and a SGKs in particular. Empirical research has been limited in scope, size, and technology form factor. Most reported text entry research has also been based on research prototypes. Continued progress and innovation in the text entry field cannot have a solid empirical footing if we do not know how well current technologies work for users. Further, despite the prevalence of STKs and SGKs there is a lack of in-depth studies about their text entry performance, in particular outside a lab environment.

While a lab experiment is the *de-facto* standard text entry evaluation methodology, we were curious to see how people use STK and SGK on their own mobile devices outside the lab amid their everyday activities. We therefore set out to conduct a text entry evaluation based on the Experience Sampling Method (ESM), in particular the way it has been used in ubiquitous computing (Consolvo & Walker, 2003). To the best of our knowledge, ESM has not been used to compare two text entry methods before. We decided to compare the text entry performance and perceived user experience of STK and SGK “in the wild” in a study in which participants performed transcription tasks whilst attending to their daily tasks. In this chapter, we empirically compare two state-of-the-art text input methods outside the lab – in a real world setting, which we call “in the wild”. This is especially interesting as we evaluate the system under a variety of circumstances, which will be described in the following sections.

4.2 Hypotheses

We present the following null hypotheses which are to be accepted or rejectd as a result of this study. As shown, this study is broad and sheds light on many different aspects of text input between the two keyboards.

H0,a There is no difference in text entry rate between the two keyboards (STK and SGK) in the wild

H0,b There is no difference in character error rate after correction between the two keyboards (STK and SGK) in the wild

H0,c The entry rate does not differ between different hand postures used, when in the wild

- STK using single index finger
- STK using single thumb
- STK using two thumbs
- SGK using single finger
- SGK using single thumb

H0,d The error rate does not differ between the different hand postures used (see H0,c), when in the wild

H0,e The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of entry rate, in the wild

H0,f The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of error rate, in the wild

H0,g When in the wild, users did not prefer one method over the other (STK or SGK)

4.3 Variables & Confounds

In this study, we identify three types of variables as independent variables, dependent variables, and confounds.

4.3.1 Independent Variables

These are the variables we explicitly control in this study.

- V1 – Keyboard Type (2 levels: STK, SGK)
- V2 – Participant (12 levels: P1-P12)
- V3 – Block (9 levels: B1-B9) – this is simply the time of the study divided into 9 equal blocks

4.3.2 Dependent Variables

These are the variables that we measure as an outcome of this study. The measurements lead to "derived dependent variables" which lead to the analysis of the study results. This means we do not measure these directly but we derive them via calculations from the dependent variables we measure. The following sub sections below describe the variables we measure vs the variables we derive.

4.3.2.1 Measured Dependent Variables

These are the dependent variables we explicitly measure.

4.3.2.1.1 Timestamp at first keystroke (T1)

Theoretically, this is when the first key is touched as the user begins to type. However, with proprietary keyboards such as GBoard, we are not provided with a call-back function when a key is entered or a gesture trail begins as the user starts to type or gesture. Therefore we use a practical alternative – in both typing and gesturing, the user first has to touch the target text field (a *TextView* in the case of Android application), to bring it to focus. This would bring up the soft keyboard and fire an *onKeyDown* event, which we can capture. Although this is not exactly when the user starts to type but slightly earlier we believe this is a good estimate of when the user begins to type as (a) the users begin to type immediately after the keyboard comes up (b) we ask the users to first "internalise" (or memorise) the stimulus phrase and then bring up the keyboard.

4.3.2.1.2 Timestamp at final keystroke (T2)

Theoretically, this is the timestamp when the user enters the last character in the sentence or phrase they intend to type. This can be captured with regards to typing on an STK by using the last *onKeyUp* event on the *TextView*, in the case of the Android platform. However, this is impossible to capture in the case of Speech input, as the speech capture interface continues to run after the user has stopped speaking, waiting for a significant spell of silence before it deactivates. Therefore when running studies, we use a practical delimiter to capture when the user has completed typing – such as pressing a button which says NEXT, or FINISHED. Realistically, in a texting mobile application this would be denoted by pressing SEND. In this study, we capture the "end of phrase" when the user indicates they want to move to the next sentence by pressing NEXT.

It is obvious that there is a slight delay between entering the last character in the response phrase and pressing next, however, this does not skew the results in the study as:

e. When typing continuously, this happens almost instantaneously
f. We explicitly tell the users to use a minimal delay between finishing typing and pressing next
g. This delay is uniform across the entire study (and does not differ much between subjects)
h. If the user does require to proofread what they typed, this should be indeed factored in to the time it takes to enter text using the given input mechanism, as this is a critical factor

4.3.2.2 Derived Dependent Variables

These are the dependent variables we calculate from the measured dependent variables. Descriptions of these can be found in Chapter 1 – Introduction, under Conventions.

- Number of characters in the response phrase (N)
- Typing duration (T)
- The Error (E)
- Entry Rate (WPM)
- Error Rate

4.3.3 Confounding Variables

These are variables that we did not try to control, but still would be consider as variables due to their confounding nature, as they can definitely affect the typing experience and performance in the study. Descriptions of these can also be found in Chapter 1 – Introduction, under Conventions.

- OOV words
- Hand Posture

4.4 Apparatus

This section explains the hardware the software apparatus used for the study.

4.4.1 Hardware

For this study, we used the participants own devices. The rationale for this is that when performing studies "in the wild" the participants must have their phone with them the whole time. If the authors were to provide participants with a mobile device, this would be unrealistic for two reasons:

(a) This would not be their primary phone – therefore collecting data via this device would not yield accurate data pertaining to their actual behaviour

(b) The participants will not be familiar with the device, therefore the data will be unrealistic

To ensure that the software apparatus runs properly, and there's not too much difference between the device form factors, we filtered the participants based on the specs of their primary mobile device. The criteria were:

(a) They must have Android 4.0 (Ice Cream Sandwich) or later

(b) Should contain Google Play Store – to download and install Google Keyboard

Upon screening and selection, the resultant devices used for the experiment were as follows (which were contemporary during the time the study was run – 2013).

Phone Make & Model	Form Factor	How many
Samsung Galaxy S3	4.8”	5
Samsung Galaxy S4	5.0”	1
Samsung Galaxy Note	5.3”	1
Samsung Galaxy S2	4.3”	1
Google Nexus 4	4.7”	1
Lenovo S720	4.5”	1
HTC Desire	3.7”	1
HTC One	4.3”	1

4.4.2 Software Apparatus

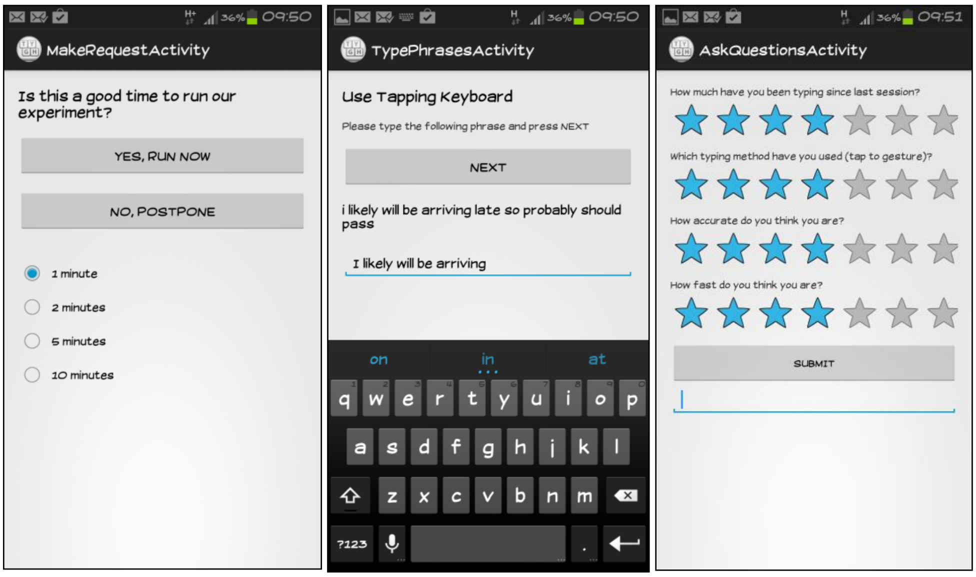
There were two major components in the software apparatus. The First was the Google Keyboard, which had its own implementation of state-of-the-art STK and a state-of-the-art SGK built in. The second was the experimental software that was required to run the study. The participants were required to download and install the Google Keyboard and set it as their main method of input, and the experimental software, described as follows.

4.4.2.1 Experimental Software

The app used for the study outlined in this chapter is designed for an ESM study carried out in the wild. It was important that the app handled everything in a fully automated manner with minimal or no experimenter intervention. The participants could find themselves in any situation during the study and the app had to be ready to deal with all these foreseen and unforeseen circumstances.

The app has a basic start up screen for entering information such as participant ID. Once started, the app will be working in the background for the duration of the experiment (e.g. 1 month). At pseudo-random intervals of the day (roughly spaced apart by 1 hour), the app will come to the foreground and request the user to perform a task. The task is a transcription task where the user is shown a stimuli sentence and requested to copy it using one of the methods (STK or SGK). However, the user mind find themselves in a situation where this is not possible, in which case they can choose to “snooze” this request for 1, 2, 5 or 10 minutes, after which the app will remind them again.

This behaviour was achieved by using the Android's AlarmManager class. If the user decides to accept the request, then the user will be shown a basic screen with a textbox and next button. The user simply has to copy the sentence given, using the condition (see image – shows "use Tapping Keyboard") and press Next. Once they do, another basic screen will show them 4 questions with 7 point Likert scales as responses (see materials), these are to capture the users perceived performance and comparative performance (in comparison to the previous session).



MakeRequestActivity

Is this a good time to run our experiment?

● 1 minute
○ 2 minutes
○ 5 minutes
○ 10 minutes

TypePhrasesActivity

Use Tapping Keyboard

Please type the following phrase and press NEXT

i likely will be arriving late so probably should pass

I likely will be arriving

AskQuestionsActivity

How much have you been typing since last session?

Which typing method have you used (tap to gesture)?

How accurate do you think you are?

How fast do you think you are?

Figure 28 - Software Apparatus used for ESM study

The stimuli phrase shown here is from a randomized copy of a given data set (we use the Enron Mobile Email Dataset for this study – see Materials), shuffled using the Fisher Yates shuffling algorithm (Fisher & Yates, 1948). The two timestamps where the user enters the first character and presses the NEXT button are also captured. The stimuli and response phrases, as well as the timestamps are written to files in the mobile's internal storage. Once this process is complete, the app goes to the background again and waits until it is time to come up.

Once the mobile connects to a WiFi network, the app identifies this and uploads the data it saved to a specified URL via a HTTP POST request. A server side application was implemented to capture this request and read the saved data from it in XML format.

The app handles any anomalies that could occur when this network transfer is taking place.

Since this app is supposed to work “in the wild”, there are so many unforeseen situations that the app must be made ready to, unlike in a lab experiment. Examples for such situations are:

- The phone can run out of battery and die. This can be between requests, or while the participant is actually servicing a request. The experiment should commence when the phone turns on again, from where it stopped, so the participant doesn’t need to redo what they had done before.

- The phone can be rotated – and due to the Android implementation, the entire android activity is killed and recreated, thus clearing all temporary and global variables stored in the app. Therefore every single action pertaining to saving state or data should always be stored in persistent storage and retrieved.

- The user may forget or ignore to provide a response at all, in which case the app must keep reminding the user. This is implemented via a tolerance duration variable (i.e. 15mins), after which the app will buzz again.

- The user might want to set the app inactive during the times he/she sleeps. This is implemented via a “quiet time” setting, where the app can be configured not to buzz a user at certain times of the day.

- The user might accidently press the home button and send the app to the background while typing, and the app should recognize this and come to the foreground again – either immediately or after the tolerance duration mentioned above. The back button has been disabled for similar reasons, but the home button cannot be disabled without rooting the device.

It should be noted here that the app does not collect any data outside these experimental sessions, thus not raising any ethical or privacy concerns for the participants. This app is distributed with the aim of aiding researchers who wish to conduct such experiments without having to "reinvent the wheel".

This application was developed using Java/Eclipse, and could be deployed on any android mobile device running Android version 4.0 or later - the requirement for version 4.0 was to align with the requirements of Google keyboard. It can be deployed as a single APK file on a mobile device with ease.

4.4.2.2 Software Design

This application contains 3 parts which can be treated as pluggable modules which interact with each other.

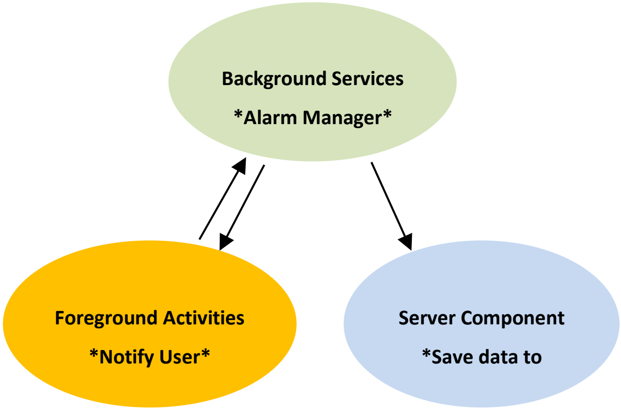


Figure 29 - High Level Component Diagram

The entire operation of the app can be simplified into the diagram below. In addition to this we built a simple server application which reads data from an HTTP request and writes it to a database. We implemented ours using PHP and MySQL, but it only requires to be a simple application that listens to Http Requests and reads XML.

All the configuration parameters are defined as constants therefore anyone could customize the application to their requirement such as the data set used (i.e Enron), number of runs (i.e. 300), number of phrases to type per run (3), delay values, I, T, P, R, etc.

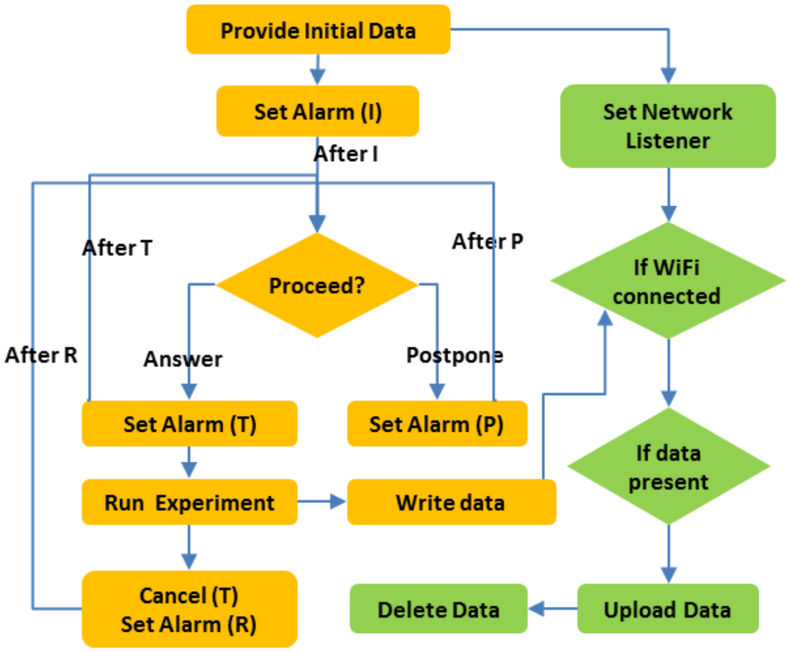


Figure 30 - Operational Flow Diagram

4.5 Materials

4.5.1 Surveys

In addition to capturing the user's performance when using either type of keyboard, we also surveyed their responses on previous typing experience, mobile phone experience, smartphone experience, and perceived performance and user experience, which shall be explained in the upcoming sub sections.

4.5.1.1 Preliminary Survey

This was given at the beginning of the entire study, before the user had the opportunity to enter any text whatsoever. The purpose of the survey was to gauge the prior experience of the user.

Q1. In your life, which text entry method did you use more during your day-to-day activity?

Only Tapping 1 2 3 4 5 6 7 Only Gesturing

Rationale (Q1):

This, in conjunction with the final survey (presented at the end of the study), this will reveal if users had prior experience with gesture keyboard, and if not, would have started using gesture keyboard in their day to day life as a result of their study.

Q2. What kind of mobile devices do you use? Tick all that apply.

Smartphone – Android
Smartphone – Apple
Smartphone – Microsoft
BlackBerry
Feature phone (no large touchscreen)
Tablet – Android
Tablet – Apple
Tablet – Microsoft
Phablet – Android (A very large smartphone)

Q3. Please write the brands and models of the mobile device you have used the most in last few years.

Brand Model Duration of Use

Q4. Please rate your ability to type using your mobile device.

Very slow typist 1 2 3 4 5 6 7 Very fast typist
Inaccurate typist 1 2 3 4 5 6 7 Very accurate typist

Rationale (Q2-Q4):

This is to gauge the user's previous smartphone experience, which is directly attributable to one's performance on a STK and SG. This will be revisited in the participants section.

4.5.1.2 Survey's during the studies

At each experience sample, we gauged the participant's "in-situ" user experience four questions, response for each of these being a Likert Scale from 1-7.

Q5. How much have you been typing since last session?

Very Little 1 2 3 4 5 6 7 All the time

Q6. Which typing method have you used outside this study?

Only Tapping 1 2 3 4 5 6 7 Only Gesturing

Rationale (Q5-Q6):

We wanted to find out how much the users have typed outside this study a.k.a. between each experience sample, just to see if that has a correlation with their performance during the sample

Q7. How accurate do you think you are?

Not at all 1 2 3 4 5 6 7 Very Accurate

Q8. How fast do you think you are?

Not at all 1 2 3 4 5 6 7 Very Fast

Rationale (Q7-Q8):

These were regards to the experience sample they just completed; we wished to find out how they perceived themselves in terms of being fast or accurate.

4.5.1.3 Final Survey – End of the study

This was the final questionnaire at the end of the full study. We required the participants to provide both quantitative and qualitative/open ended answers on what they liked and disliked about each input method. These questions were based on the entire study experience a.k.a. the full 1 month duration.

Q9. During this study, I was mostly

Stationary 1 2 3 4 5 6 7 On the move

Q10. During the study, I was mostly

Energetic 1 2 3 4 5 6 7 Exhausted

Q11. I most actively interact with my mobile device during

Morning 1 2 3 4 5 6 7 Night

Q12. My general method of text input before the experiment was

Tapping 1 1 2 3 4 5 6 7 Gesture

Q13. My general method of text input after the experiment is

Tapping 1 2 3 4 5 6 7 Gesture

Q14. When the app made a request when I was walking, when entering text I

Stopped 1 2 3 4 5 6 7 Continued walking

Rationale (Q9, Q14):

As the question suggests, we wanted to find out if the users were mostly stationary or on the move during the study. This is simply to obtain an idea of what the circumstances we sampled their experience in.

Rationale (Q10-Q11):

We wanted to find out if fatigue, energy levels, concentration, or time of the day can affect typing performance with relation to our participants during the study.

Rationale (Q12-13):

This was to find out if the study had actually affected the users day-to-day typing. As some users had not been exposure to gesture keyboard before this study, we wanted to find out if they used it regularly.

Q15. What did you like about each input method?

Q16. What did you dislike about each input method?

Q17. Did your everyday text input get affected as a result of this experiment?
Did you learn a new method of text input (i.e. gesture input), became aware about a new tool (i.e. Google keyboard), apply it to your own day-to-day life, or did you become faster, more accurate etc.

Q18. What features in the application did you find desirable?

Q19. What features in the application did you find un-desirable?

Q20. What improvements would you suggest we do for this app if we plan to run the study again?

Q21. What do you think about this experiment?

Q22. What was the hand posture that you used for each typing method (pick the most used).
TAPPING – Thumb | Two Thumbs | Single Finger
GESTURE – Thumb | Two Thumbs | Single Finger

Rationale (Q15, Q16):

We wanted to capture which aspects of each input method they like and disliked, as this would give us insight into what features work better and when

Rationale (Q17):

This was to strengthen and justify the values in Q12 and Q13

Rationale (Q18-Q21):

This was simply to find out what could be improved with the study. We did not use this information in the results part of this thesis, yet the suggestions made here were used to improve the studies (see Chapter 6)

Rationale (Q22):

We wanted to gauge the participants hand posture during this study when using STK and SGK

4.5.2 Phrase Set

We used a subset of the Enron mobile email dataset (Keith Vertanen & Kristensson, 2011) with the following conditions:

- Each sentence should be less than 60 characters in length
- No numbers
- No special symbols

This resulted in phrase set of 1008 phrases. We counted 1,457 unique words in this test set. The rationale behind this was we didn’t want users to switch between keyboards to enter numbers and special symbols – i.e. in Android, when using Google Keyboard; users have to change the view back and forth to enter numbers, symbols and letters. We decided this should be explored in a different study than this one.

4.5.2.1 OOV Words

We did not, however, exclude sentences with Out of Vocabulary words (OOVs). We did this because we want to find out how each keyboard type (STK, SGK) performed differently when OOV words were part of the mix. As SGK had no way of inferring OOV words, it was an interesting observation as to how it affected the typing speed, the error rate, the user experience and most of all how users dealt with this particular issue when typing.

We compared all the words in the phrase set against a standard lexicon (64K common words used in the English language). The words that weren’t in the lexicon were each entered carefully on the Google keyboard, by tapping the center of each key on the STK and by gesturing from the center to center of each key on the SGK. We noted that the same 44 words were out of vocabulary (OOV) words for both the STK and SGK. These OOVs appeared in 45 sentences (4.46% of 1,008) in the phrase set, and were marked as sentences with OOV words. These OOV sentences were analyzed in post-hoc analyses after the experiment.

4.5.2.2 Non OOV Samples

The following are a few sample phrases from the set which do not include OOV words

“I have received your messages and will respond accordingly”

“Please make sure Bob Kelly is on the list”

“I was answering Janet's comment”

“Anything exciting going on today”

“Email the consent to me”

4.5.2.3 OOV Samples

The following are a few sample phrases from the set which include OOV words

“Check with Vince Strohmeyer”

“If so Whitt is done”

“Why don't you ask Shanna Funkhouser for the details”

“It's death or dynegy with no clear leader”

“Are Linde and Kim available to assist rod”

4.5.2.4 Ordering of Phrases

The Fisher Yates Shuffling Algorithm (Fisher & Yates, 1948) was used to randomize the order of the phrases when presented to the participants. The algorithm is a simple, in place shuffling mechanism and can be outlined as follows using the Java programming language.

```
public static void fisherYatesShuffle(int[] ar) {
    // generate a randomizer
    Random rnd = ThreadLocalRandom.current();

    for (int i = ar.length - 1; i > 0; i--) {
      // generate random index between 0 and (i+1)
      int index = rnd.nextInt(i + 1);

      // perform a simple swap between the current and
      // random positions
      int a = ar[index];
      ar[index] = ar[i];
      ar[i] = a;
    }
}
```

4.6 Compensation

Participants were compensated £50 for their time in amazon vouchers. Further we offered an incentive of an extra £15 for the fastest typing participant in each keyboard type, under a certain error rate threshold.

4.7 Participants

We recruited 12 volunteers from the university campus. Again these too were a rather broad sample as they came from various schools and departments. None of the participants in the ESM study had participated in the lab-based study.

4.7.1 Participant Demographics

Due to the ethics agreement we cannot publish any identifiable information about the participants - therefore the aggregate results of each demographic will be described below and not attributed to individual participants. Again, as the lab based study, the participant number was justified by previous studies performed in literature (P. O. Kristensson & Denby, 2009)

4.7.1.1 Gender

This time we had 7 males and 5 females.

4.7.1.2 Age

Their ages ranged from 21 to 42, with a mean of 27 and Standard Deviation of 6. This again ensured we had a satisfactory distribution of ages which is quite representative of the real world population who uses smartphones for text entry in the year 2013 (when this experiment was performed).

4.7.1.3 English Proficiency

Three of them had English as their first language whilst the others practiced English as their second language. As per the previous experiment the participants used English regularly for studies and conversation. Given they were all doing either a undergraduate, postgraduate or PhD in University of St Andrews they had to be proficient in English if not they would not be admitted for study – as per the English language requirements of university admissions. This ensured that our participants were able to understand, read and copy the sentences in the above phrase set without difficulty. Most of the participants used the English language in their day to day life for exchange of messages over mobile devices except for two of them who were using a keyboard with English keys but output transliterated Chinese characters. They were screened for their competency in English and proved to be satisfactory.

4.7.1.4 Geographic Distribution

The best part about running studies in University of St Andrews is that it attracts students from all over the world. In a recent survey, it was found that St Andrews students represent 120 different cultures, which gives a mini sample of the global population. Our study therefore, had participants from 10 different countries. The 12 participants were distributed across the globe as follows.

Russia	Thailand	Scotland	England	Nigeria	Japan	Hong Kong	India	Saudi Arabia	China
1	1	1	1	1	1	1	2	1	2

4.7.1.5 Field of Study Distribution

The participant’s field of study also varied across the disciplines – this also ensures that our participant distribution was satisfactory in terms of different levels of technical expertise The participant’s fields of study can be summarised as follows.

Computer Science	Other Sciences	Arts & Humanities
6	4	2

Further, the participants’ level of study can be summarised as follows.

Undergraduate 2nd Year	Undergraduate 4th Year	Masters	PhD
1	1	4	6

4.7.1.6 Smartphone Experience

All the participants had experience with Android, and experience with Google Keyboard, as we chose participants who only owned Android mobile devices running version 4.0 (Ice Cream Sandwich) or later.

4.7.1.7 Exposure to STK & SGK

Again by interviewing the participants, we gauged their previous experience with STK and SGK, which is probably the most relevant demographic survey related to this study.

Experience with QWERTY	Experience with STK	Experience with SGK
all	all	4

4.7.2 Design

We incorporated a within-subjects design for this study - meaning each participant had to experience both conditions (STK and SGK). To minimise starting bias, we had 6 participants use STK first, and the remaining 6 participants to use SGK first. This ensured that we had a balanced group of participants with fully balanced conditions between the two groups. The participants were allocated randomly to the two groups.

4.7.3 Recruitment

We used various channels to advertise this study and recruit participants, briefly outlined below. A sample advert used for this study is shown in the image below.

4.7.3.1 Screening

Participants were screened for:

(d) English proficiency – either they had to be native English speakers or use English as a second language in their day to day life / studies

(e) Experience with QWERTY – since we use this keyboard layout, participants had to be experienced in typing using a QWERTY keyboard either using the computer or their mobile device

(f) Type of mobile device – they had to own an Android mobile device running version 4.0 or later, and had to have the Google Play Store available

Not all applicants were recruited as some of them were filtered based on the compatibility of their mobile device. We ensured that we recruited participants with a standard android 4.0 or above installation and a suitable mobile device where the only method of input would be a software keyboard via a touch screen (no devices with built in hardware keyboards in them). I.e. we didn’t recruit an applicant who had a non-standard Samsung Galaxy S4, and was unable to install Google keyboard as it did not have a Google Play Store instance built into the operating system.

Participants needed for a study on Text-Input.
Compensated £50 + Chance to win extra £15 in Amazon Vouchers.

We invite you to participate in a PhD research project examining how typing speed and error rate changes when entering text on an Android mobile device with two different input methods (Tapping Vs. Gesturing). We will be using the Google keyboard which supports normal tapping of keys on a virtual (soft) keyboard and also continuous input for gestures.

The only requirements for participation are that you are above 18, used to entering English text on a mobile device, do not suffer from any learning or communication disabilities, and own an android mobile device.

Each participant will be required to install Google keyboard (if not already present) and a custom application on their mobile device. The application will run in the background for 4 weeks. The application will come to the foreground at random times of the day (~10 occurrences a day) and ask the user to enter 3 phrases. The app will record the typed phrase, input speed and error rate, as well as motion sensor data from your mobile to identify whether the participant was still or moving.

When the user connects to wifi network, the aforesaid recorded data will be sent back to a server where it will be collected. The application will not record any other data while running in the background.

You will be reimbursed £50 for your time, provided that you follow the experiment instructions. Two participants who fare the highest speed and accuracy in each method of input will be awarded an extra £15 each. All payments will be made in the form of Amazon Vouchers.

If you are interested in participating please contact Shyam Reyal on
smr20@st-andrews.ac.uk or 07447924147.

4.8 Procedure and Execution

The execution of the study was done in two steps. The first was a pilot where the authors of the paper used the apparatus to find any errors in the implementation that could affect the study, followed by the actual study involving the 12 participants.

4.8.1 Pilot Study

The study apparatus was piloted intensely before the actual study commenced, and a number of problems were identified. The main problems found during piloting were:

- Users completely ignoring the request, they do not respond, nor snooze the request – this is when the "tolerance duration" was introduced to remind them again
- The app did not work as expected with different form factors i.e. the HTC Desire with a 3.7" screen, therefore certain buttons and textboxes had to be resized
- One of the piloting users complained that the app was using mobile data, after which we decided to save the data to the mobile upon sampling each experience point, and then upload the data only upon finding a WIFI connection
- When the phone was turned off and back on, the application did not come up after the required time – therefore a "Boot Receiver" was implemented to trigger the application whenever the phone was restarted
- Anyone would have been able to upload fake data to the server, thus messing up the results. Therefore an authentication system was implemented in the server such that only files from the enrolled participants phones will be allowed for upload

4.8.2 Actual Study

This procedure was fairly straightforward. The participants were given a custom built application to install on their mobile phones. The app ran in the background and from time to time asked the users to enter three phrases from the modified Enron mobile email data set (same as used for the longitudinal study). The users had the option of either responding immediately or postponing the request by either 1, 2, 5 or 10 minutes. If chosen to postpone, the app would remind them again in the designated time.

The experiment used a transcription task where participants were shown a phrase from the dataset and asked to copy it. Each participant was given a mobile application to be used over duration of 4-5 weeks, on which they would enter text during random times of the day. Each participant was compensated with £50 for their commitment in the form of amazon vouchers. To encourage the participants to enter text as quickly and accurately as possible, a reward of extra £15 was announced to the participants who fare the fastest and most accurate.

Whilst being encouraged to use the Google keyboard’s suggested words for correction, we discouraged participants to go back and correct errors unless absolutely necessary. I.e. when transcribing the phrase, if an error or typo was made, we asked the participants to decide if they would send this message off if a human user was listening on the other end. If they decided the original word was still able to be deciphered by the recipient, then they were not required to go back and correct it. Our experiment app recorded the stimulus phrases and the response text using millisecond timestamps when the user entered the first character and when the user pressed NEXT. Participants rated their previous experience with software keyboards (STK and SGK) on mobile devices, and self-rated themselves on how fast and accurate they thought they were.

If chosen to responds immediately, the app would display the sentence and once the user starts entering text would record the time between when the first character was entered and when the user pressed NEXT. It should be noted here that the app did not collect any data outside these experimental sessions.

The app was configured to give each participant 300 tasks over the full duration of the experiment, which was about 10 tasks per day, evenly spread during times the participants could be expected to be awake (the exact times were determined specifically for each individual study participant). Each sample required users to transcribe three phrases, thus collecting around 900 data points from each participant. We used the same Enron mobile phrase set as in the previous study.

We had timed the application so that the user would get around 10 requests a day at random time slots which are nearly equally spaced out during the time of day which the user is awake and is capable of interacting with his/her mobile device. Once the user has entered three sentences, the app goes to the background following a very brief questionnaire where the user has to reflect on his/her overall daily typing experience. We asked the user how much they typed, and to rate themselves for accuracy and speed, and also which method they used mostly STK or SGK. These were to be rated on a scale of 1-7 as usual.

The goals were to capture text input performance in a variety of everyday environments and mobility settings. The participants could defer a sampling prompt if it were inopportune. In practice they accepted prompts when they were standing, walking, using the computer, during lectures, while cooking, while travelling in moving vehicles and whilst lying on the bed.

Half of the participants used STK for the first two weeks and half of them used SGK. After two weeks the participants switched to the other text entry method.

We intentionally did not control hand posture. Instead we asked participants to use their preferred posture and report it at the end of the study. The choices were single thumb, single finger and two thumbs. At the end, participants were asked to write descriptive and open comments about what they liked and/or disliked about each text entry method.

4.9 Results & Analysis

Each participant entered an average of 469 (SD = 13) phrases on STK, and 447 (SD = 45) phrases on SGK. This resulted in 5,623 and 5,363 data points for STK and SGK respectively. We discarded 97 and 120 data points as outliers based on the same filtering criteria as in the previous lab study. After filtering we ended up with 5,526 and

5,243 valid data points for STK and SGK. We split these data points into nine blocks, such that each block contained around 50 ordered data points.

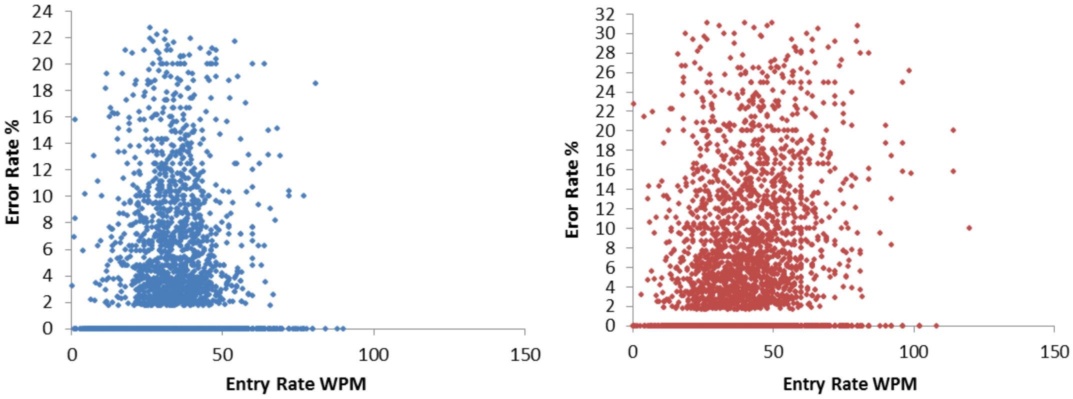


Figure 31 - Entry Rate vs Error Rate Scatter Plots for STK and SGK in Experiment B

4.9.1 Entry Rate

SGK was significantly faster than STK, and participants improved more with SGK than with STK with practice.

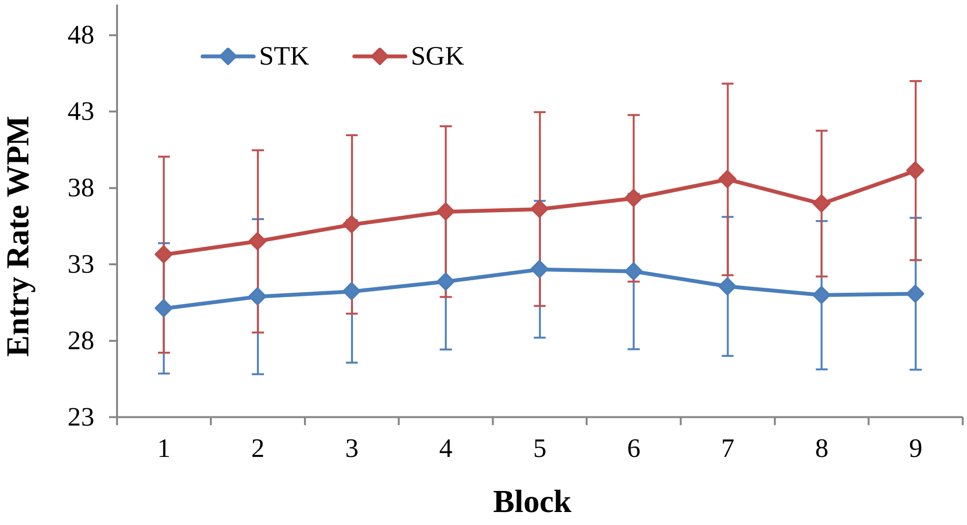


Figure 32 - Entry Rate vs Session for STK and SGK in Experiment B

There is a striking difference in speed patterns between Experiment 1 (Chapter 3) and Experiment 2 (Chapter 4). While the STK results are quite similar, the SGK results in the ESM experiment started much faster and grew even higher in speed as the study progressed.

WPM	*m* S1	95% CI S1	*m* S9	95% CI S9
STK	30.1	25.9 – 34.4	31.1	26.1 – 36
SGK	33.6	27.2 – 40	39.1	33.3 – 45
ANOVA	F_{df}	df	η_p^2	p
Input Type	5.965	1,11	.352	**.033***
Session	3.818	4,44	.258	**.001***
Input × Session	1.094	4,44	.090	.375

4.9.2 Character Error Rate

SGK produced significantly higher CER than STK, which was similar to the lab study in Chapter 3, but with higher values in both conditions.

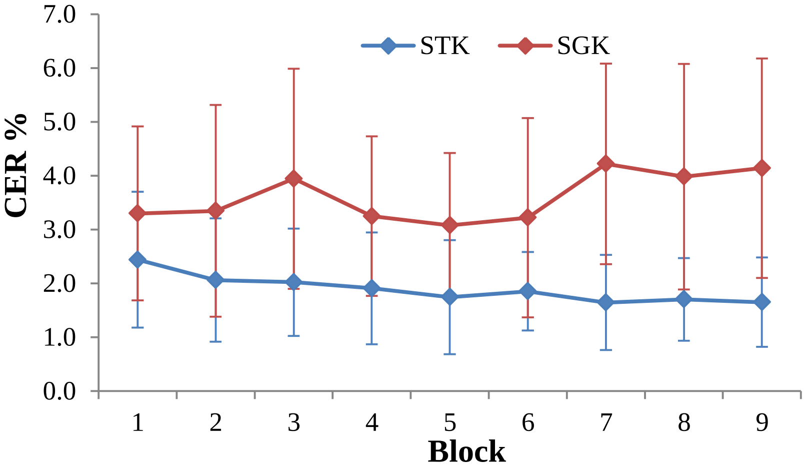


Figure 33 – CER vs Session for STK and SGK in Experiment B

CER%	*m* S1	95% CI S1	*m* S9	95% CI S9
STK	2.44	1.18 – 3.71	1.65	0.82 – 2.48
SGK	3.30	1.69 – 4.92	4.14	2.1 – 6.18
ANOVA	F_{df}	df	η_p^2	p
Input Type	10.552	1,11	.490	**.008***
Session	1.375	4,44	.111	.219
Input × Session	1.559	4,44	.124	.149

4.9.3 Word Error Rate

Word Error Rate followed a similar pattern to Character Error Rate, but had higher values.

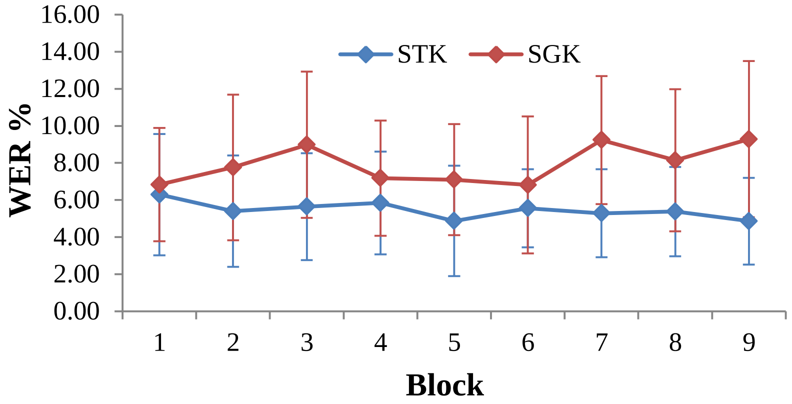


Figure 34 - WER vs Session for STK and SGK in Experiment B

4.9.4 Excluding Sentences with OOV Words

As in the lab study in Chapter 3, we excluded 465 (4.31% of 10,769) data points which contained the identified OOV sentences.

4.9.4.1 Entry Rate

SGK was still significantly faster, produced significantly more errors, and participants improved over the sessions.

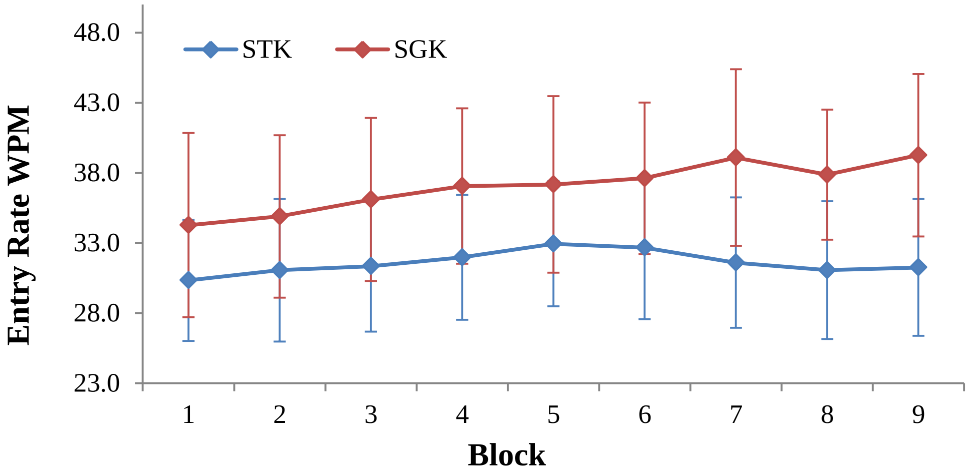


Figure 35 - Entry Rate vs Session for STK and SGK in Experiment B – excluding OOV words

WPM	*m* S1	95% CI S1	*m* S9	95% CI S9
STK	30.3	26 – 34.7	31.3	26.4 – 36.1
SGK	34.3	27.7 – 40.8	39.3	33.5 – 45
ANOVA	F_{df}	df	η_p^2	p
Input Type	7.015	1,11	.389	**.023***
Session	3.376	4,44	.235	**.002***
Input × Session	1.083	4,44	.090	.382

4.9.4.2 Character Error Rate

The character error rate followed a similar pattern to the previous character error rate distribution. The difference between the results was significant.

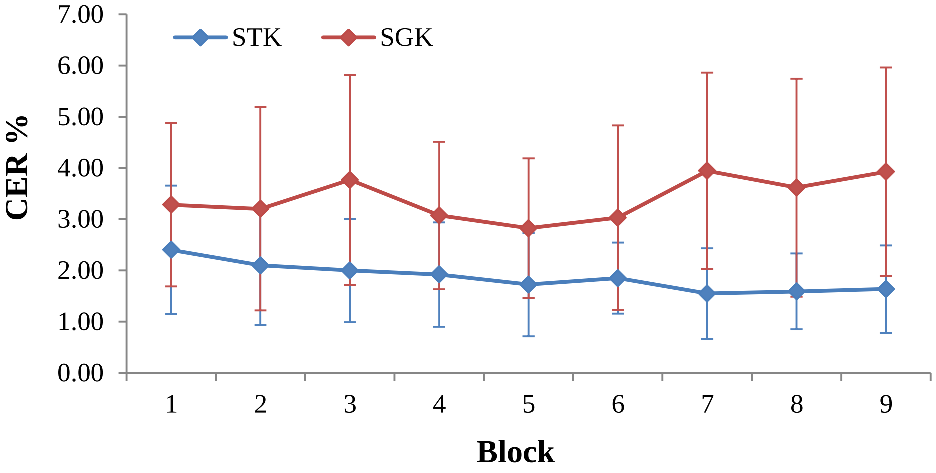


Figure 36 – CER vs Session for STK and SGK in Experiment B – excluding OOV words

CER	*m* S1	95% CI S1	*m* S9	95% CI S9
STK	2.40	1.15 – 3.66	1.64	0.78 – 2.49
SGK	3.29	1.69 – 4.88	3.93	1.89 – 5.96
ANOVA	F_{df}	df	η_p^2	p
Input Type	8.298	1,11	.430	**.015***
Session	1.329	4,44	.108	.240
Input × Session	1.403	4,44	.113	.206

4.9.4.3 Word Error Rate

Word error rate followed a similar pattern to the character error rate with higher values.

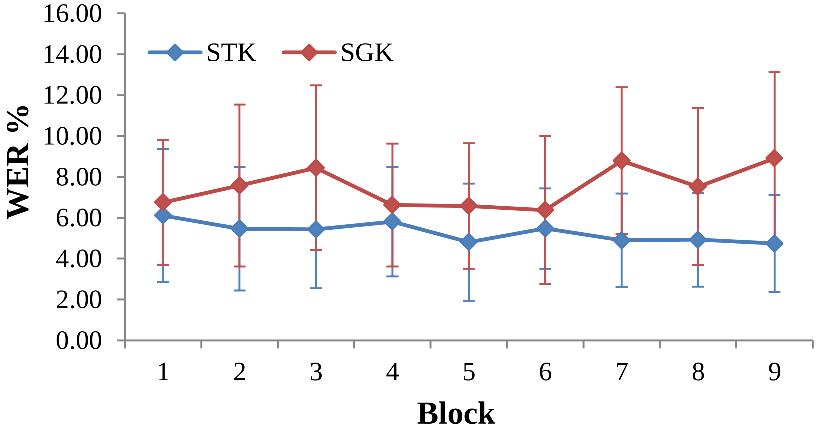


Figure 37 - WER vs Session for STK and SGK in Experiment B – excluding OOV words

4.9.5 Considering sentences only with OOV words

As in the lab study in Chapter 3, we investigated those sentences containing OOV words.

4.9.5.1 Entry Rate

This was particularly interesting as the entry rate for SGK dropped so low that it was no longer faster than STK as before This is also similar to what we noted in the lab study; OOV's greatly impact the entry rate of SGK.

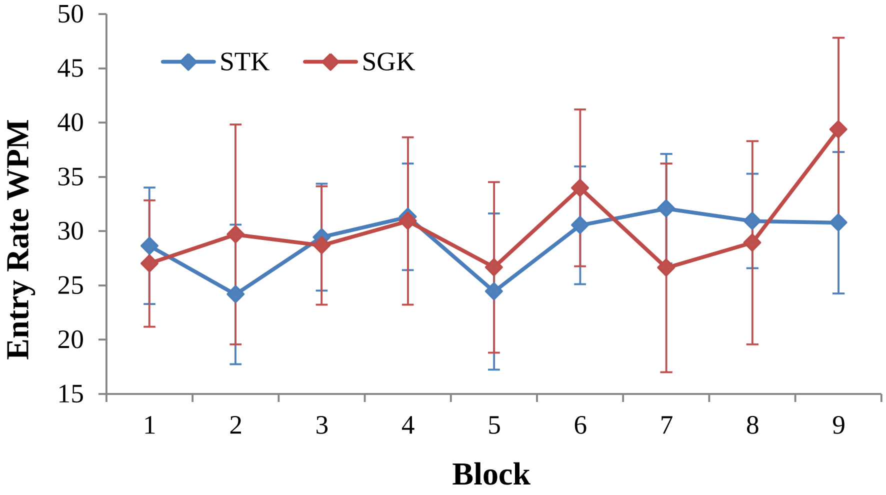


Figure 38 - Entry Rate vs Session for STK and SGK in Experiment B – considering only OOV sentences

WPM	*m* S1	95% CI S1	*m* S9	95% CI S9
STK	28.6	23.3 – 34	30.8	24.3 – 37.3
SGK	27.0	21.2 – 32.8	39.4	30.9 – 47.8
ANOVA	F_{df}	df	η_p^2	p
Input Type	0.135	1,11	.012	.720
Session	2.822	4,44	.205	**.008***
Input × Session	1.588	4,44	.126	.140

4.9.5.2 Character Error Rate

SGK produced much higher CER than STK and there was a significant change in the error rates across the blocks.

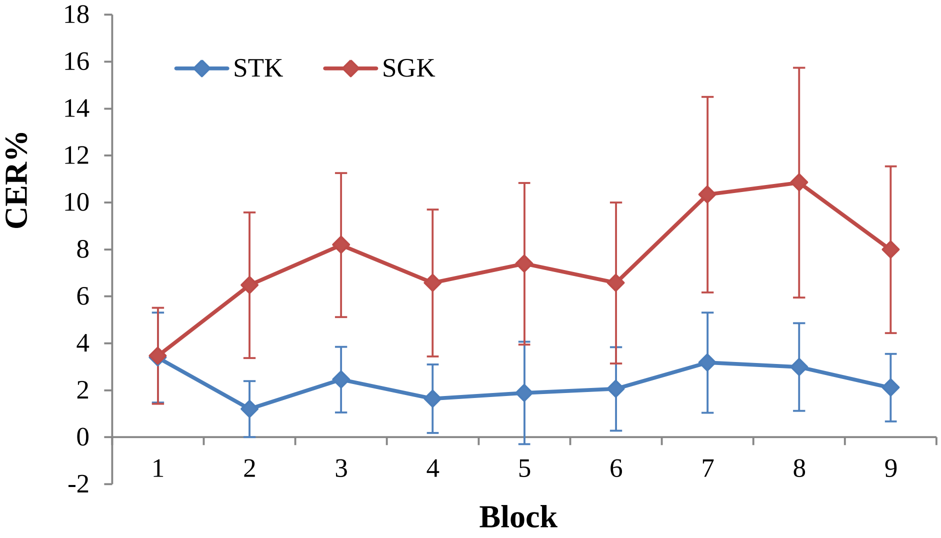


Figure 39 - CER vs Session for STK and SGK in Experiment B – considering only OOV sentences

CER	*m* S1	95% CI S1	*m* S9	95% CI S9
STK	3.39	1.48 – 5.31	2.11	0.67 – 3.55
SGK	3.46	1.42 – 5.51	7.98	4.43 – 11.54
ANOVA	F_{df}	df	η_p^2	p
Input Type	51.018	1,11	.831	**.000***
Session	2.718	4,44	.198	**.010***
Input × Session	1.635	4,44	.129	.126

4.9.5.3 Word Error Rate

Word error rates also followed a similar pattern to character error rates.

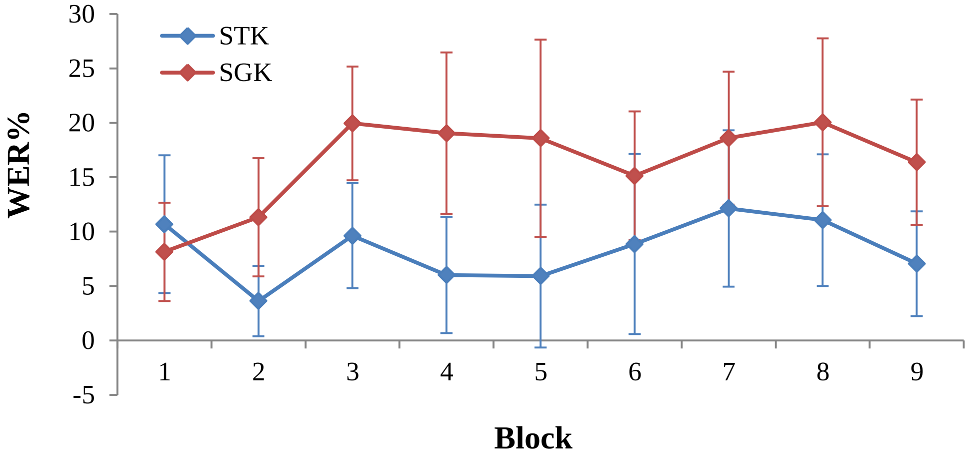


Figure 40 - WER vs Session for STK and SGK in Experiment B – considering only OOV sentences

4.9.6 Analysis of Hand Postures

When using STK, eight participants mostly used two thumbs to type (3,686 data points) and four participants mostly used a single finger (1,843 data points). When using SGK, nine participants used single finger (4,057 data points) while three users used single thumb (1,186 data points). No participants opted for neither bi-manual gesture input on SGK or single thumb on STK. The number of participants who used the same hand posture is quite similar in both the studies; therefore we can quantitatively compare the results. In the ESM study, single thumb SGK was the fastest, followed by single finger SGK and two-thumb STK.

4.9.6.1 Entry Rate

Compared to Experiment 1 (Chapter 3) it can be seen immediately that the entry rate is lower for STK and higher for SGK. Two-thumb STK is not the fastest text entry method/posture in Experiment 2, a position taken over by single-thumb-SGK. Also, while the single finger input was faster in SGK in Experiment 1, single thumb is faster in Experiment 2. However, within STK, the two-thumb posture still outperforms STK with a single finger. Two-thumb STK produced the lowest CER, followed by single thumb SGK. As in Experiment 1, the hand posture results should be interpreted as indicative.

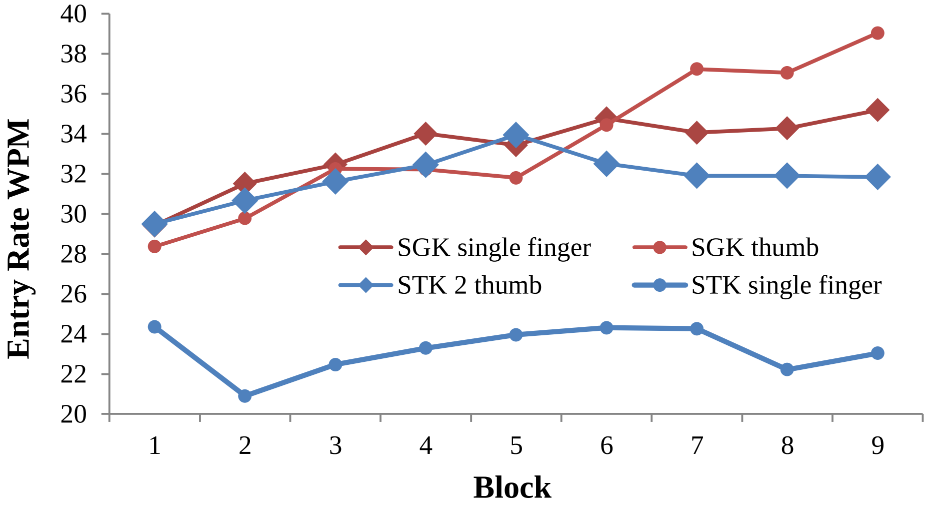


Figure 41 - Entry Rate vs Session for Hand Postures in Experiment B

4.9.6.2 Character Error Rate

When comparing error rates across the two experiments, the ESM study has higher values. The two-thumb STK produces the lowest error rate across both studies. In

Experiment 1, within SGK, single finger input resulted in a lower error rate, but in Experiment 2 this position is taken over by single thumb. Also, in Experiment 1, all STK hand postures had lower error rates than SGK hand postures, but in Experiment 2 the ranking is more mixed.

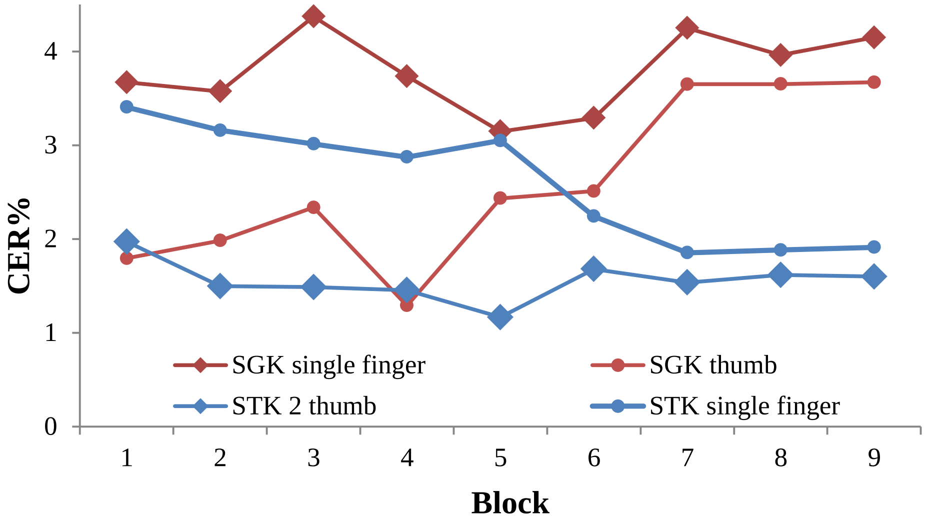


Figure 42 - CER vs Session for Hand Postures in Experiment B

4.9.6.3 Word Error Rate

Word error rate followed a similar pattern to the character error rates but with higher values as expected.

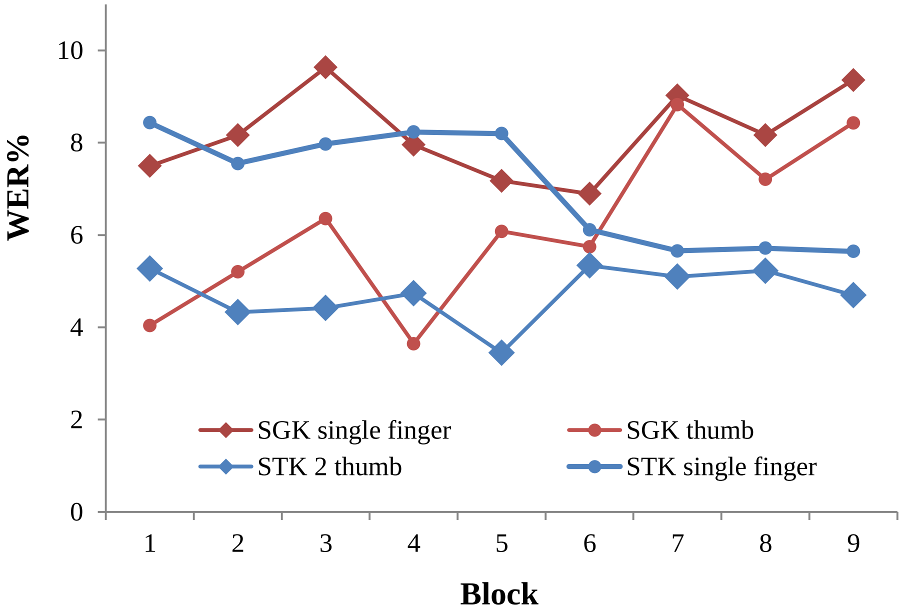


Figure 43 - WER vs Session for Hand Postures in Experiment B

4.9.7 Subjective Ratings

The participants provided ratings on how the study affected their regular text input practices. At the beginning and end of the study, we collected ratings about the users' SGK usage level outside the experiment; with 1 meaning they only used STK, and 7 meaning they only used SGK.

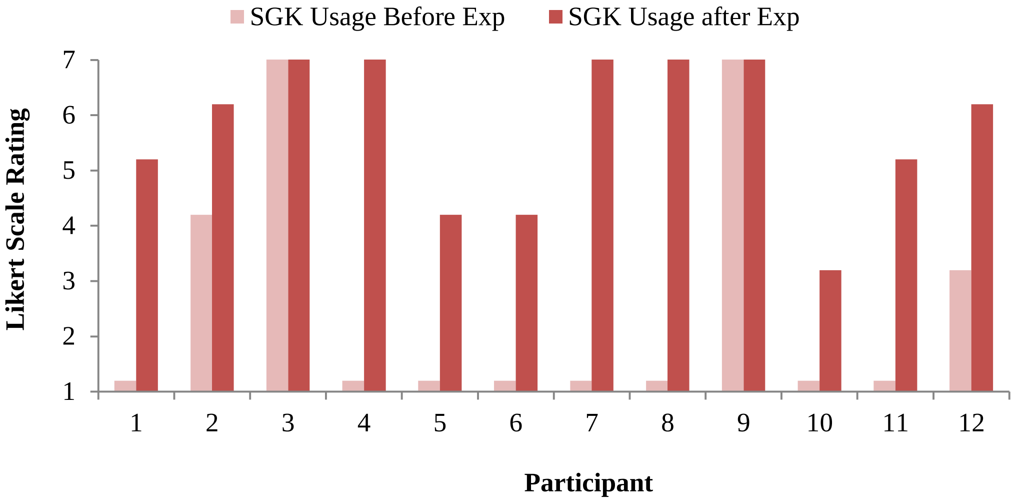


Figure 44 – SGK usage of participants in their day to day life, before and after Experiment B

This shows that at the beginning of the study eight participants were not using SGK outside the experiment at all. But at the end of the study, three participants within that set of eight participants had completely converted to using SGK as their main text entry method outside the experiment, with the other five participants reaching at least a halfway point (50/50 usage of STK and SGK). Two users have always used SGK, and the study had not affected their preference. Two other users had used both STK and SGK at the beginning of the experiment, and towards the end they had also shown a shift towards using the SGK more. It is remarkable that all users who only used STK prior to participating in the experiment were affected by this study.

The next figure shows the users' subjective ratings on their experience during the random times they were requested to participate in the study. The top two plots show that some participants were on the move when the ESM app requested them to type on their phones, and some of them continued to walk while typing instead of stopping. The bottom two plots show how busy, distracted or tired they were when actually typing texts. This is particularly useful in understanding how different the environment was when compared to the lab based Experiment 1, where the users were seated in a quiet environment, rested, and fully focused on the experiment task.

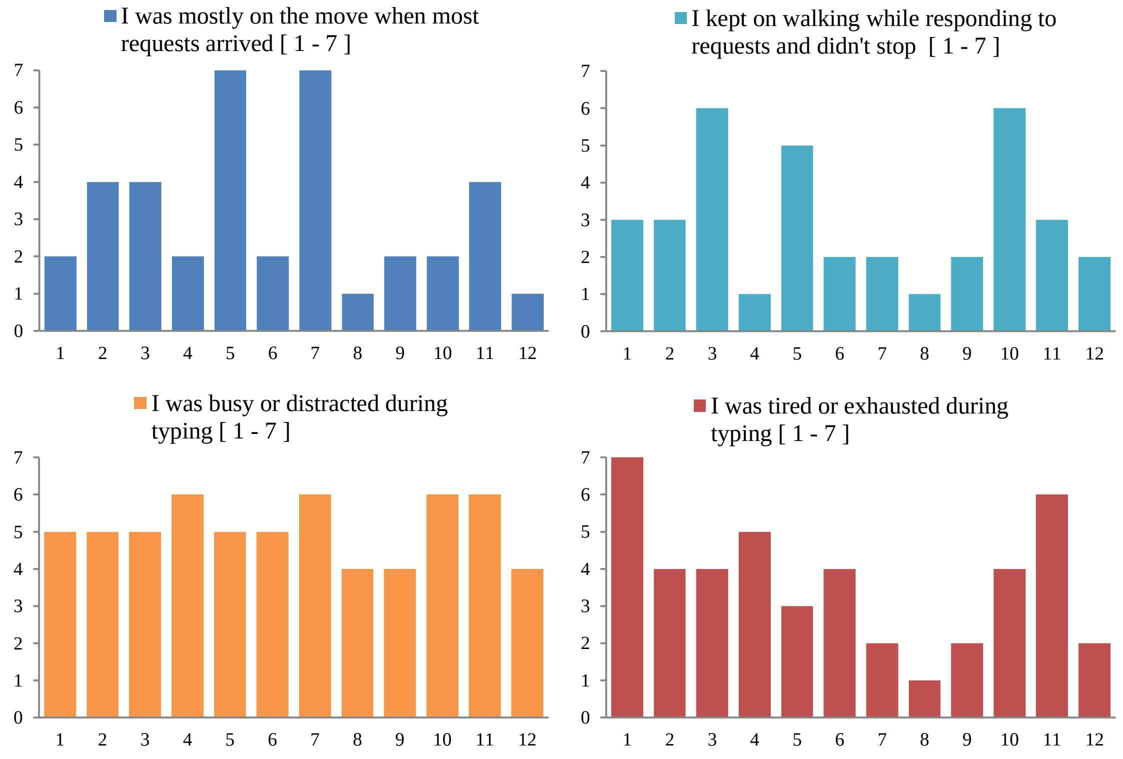


Figure 45 – Meta Data about users movement and distraction levels at experience sampling points

4.9.8 Open Comments

A few representative comments for and against each text entry method are as shown below. Participants provided general comments as well as what they liked and disliked about each method of input.

4.9.8.1 General Comments

1. “I learned to use Gesture Keyboard and I'm currently using it. I also learnt about Google keyboard and is now my default method of input “

2. “Yes I will use gesture more “
3. “I will use gesture as much as I can for typing and also will introduce it to my friends “
4. “It’s the first time I learnt this method and will definitely continue using it over tapping for my everyday use “

5. “I will use gesture input on the samsung keyboard (galaxy s4 user) “
6. “I will continue to use Google keyboard, both its tapping and gesture functions feel more user friendly than the samsung keyboard (galaxy s2 user) “

7. “I like samsung keyboard better” - (galaxy note user)

4.9.8.2 Dislike about SGK

8. “thumb gets tired quickly”

9. “limited to words provided by the dictionary, have to go back and tap to get the required words”
10. “sometimes breaks a single word into two”
11. “not good with names and proper nouns”
12. “cannot correct one or two letters, have to start from the beginning”
13. “when typing a really long word, I sometimes get confused and move in the wrong direction”

4.9.8.3 Dislike about STK

14. “difficult to have both speed and accuracy”
15. “felt clumsy”
16. “difficult to type while in motion, moving or walking”
17. “keys are quite small and very easy to make mistakes”
18. “never could type as fast as the QWERTY on blackberry”
19. “boring”

4.9.8.4 Like about SGK

20. “It’s fun”
21. “easier to use with one hand”
22. “feels natural and takes less effort”
23. “more accurate than typing”
24. “automatically adds a space to my words”
25. “you don't have to be very accurate as it can recognize the intended word”
26. “people can learn and get used to it easily”
27. “is rarely frustrating”

4.9.8.5 Like about STK

28. “ability to type words not in the dictionary, which are sometimes necessary”
29. “not restricted by the words suggested by autocorrect”
30. “ability to correct mistakes instantly”
31. “More flexibility over what I type”
32. “haptic feedback is also useful as a way of confirming you typed the correct number of letters”
33. “could use two thumbs instead of one so one hand doesn't get tired”

4.9.8.6 Analysis of Open Comments

We can see from the General comments (1,2,3,4,5,6) that users were willing to adopt gesture keyboard as their regular form of typing in their everyday lives. This is also indicative of the adaptability of the gesture keyboard a successful method of input.

Also most of the comments which the users “like about STK” and “dislike about SGK” are again to do with entering proper nouns and OOV words, which is similar to experiment 1 (chapter 3). We could see this in comments 28, 29, 30, 31, and 9, 10, 11, 12.

When it comes to general user experience, SGK seemed to be the preferred method as we can see in comments 20, 21, 22, 23, 25, 26, 27 in “like about SGK” and 15, 16, 17, 19 on “dislike in STK”. From this it’s again indicative that STK is preferred for OOV words and SGK has a general method of entry.

4.10 Summary of Results & Analyses

The following is a summary of the quantitative analysis performed on the results of this study.

4.10.1 Entry Rates

In this table, we report the entry rates from the various analyses performed.

Mean Entry Rates	Normal	Excluding OOV Words	Only Sentences with OOV Words
STK	33.9 (SD=11.8)	34.1 (SD=11.9)	31.4 (SD=13.7)
SGK	40.5 (SD=15.5)	40.9 (SD=15.5)	30.5 (SD=9.6)
Is difference significant?	Yes	Yes	No
Improvement over the sessions?	Yes	Yes	Yes

So this tells us that, in the wild, under a normal phrase set mixed between OOV and non-OOV words, SGK outperforms STK in terms of entry rate. Further, participant's entry rate will improve with time. Also, we can see that the presence of OOV words impacts SGK more – i.e. when considering only sentences with OOV word, the SGK entry rate drops so low that the difference is no longer significant.

Therefore we have enough evidence to reject the following null hypotheses:

H0,a There is no difference in text entry rate between the two keyboards (STK and SGK) in the wild

H0,e The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of entry rate, in the wild

And replace them with the following alternate hypotheses:

H1,a SGK has a faster entry rate than STK in the wild

H1,e The use of Out of Vocabulary (OOV) words affect SGK more than STK differently in terms of entry rate, in the wild

4.10.2 Error Rates

In this table, we will consider only Character Error Rates (CER) as this was the basis for the statistical analysis. Word Error Rates were observed to follow a similar pattern with higher values.

Mean Error Rates	Normal	Excluding OOV Words	Only Sentences with OOV Words
STK	1.76% (SD=3.88)	1.72% (SD=3.87)	2.59% (SD=4.04)
SGK	3.36% (SD=6.04)	3.18% (SD=5.95)	7.55% (SD=6.98)
Is difference significant?	Yes	Yes	Yes
Improvement over the sessions?	No	No	Yes

So this tells us that, in the wild, under a normal phrase set mixed between OOV and non-OOV words, STK outperforms SGK in terms of error rate. Further, participant's error rate did not change with time for sentences without OOV words. When using sentences with OOV words, the participant's error rate actually increased with time.

Therefore we have enough evidence to reject the following null hypothesis:

H0,b There is no difference in character error rate after correction between the two keyboards (STK and SGK) in the wild

And replace them with the following alternate hypothesis:

H1,b SGK produces more errors than STK in the wild

However, we do not have enough evidence to reject the following null hypothesis, as with and without OOV words, there is still a significant difference in error rates between the two input methods.

H0,f The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of error rate, in the wild

4.10.3 Hand Postures

In this table, we summarise the results from the hand posture analysis.

Mean Entry Rates	Two Thumb	Single Finger	Single Thumb
STK	36.02 (SD=11.5)	-	29.77 (SD=11.39)
SGK	-	38.59 (SD=13)	41.00 (SD=16.1)

Table 1. Statistical Table – Entry Rate vs Block in Study B without OOV words

Mean Error Rates	Two Thumb	Single Finger	Single Thumb
STK	1.43% (SD=3.42)	-	2.43% (SD=4.59)
SGK	-	2.46% (SD=4.93)	3.63% (SD=6.31)

These results are indicative only, as a) hand postures were not controlled in the experiment, b) hand postures were self-reported by the participants, and c) some participants varied their hand postures across sessions. For these reasons, we do not report results of statistical analyses for hand postures. Although, the data suggest hand posture might be an important factor for complete understanding of STK and SGK performance and we could conjecture the following (as indicative only):

- Hand postures might have different effects on STK and SGK, in the wild.
- In the wild, it seems that single thumb SGK outperforms all other hand postures, with single thumb-STK being the slowest
- In the wild, the two-thumb SGK (or Bi-Manual SGK), and single finger for STK was probably not popular choices among users

However, given the non-controlled nature of this variable, the self-reporting, and not having statistical justification, this does not give us enough evidence to reject the null hypotheses:

H0,c The entry rate does not differ between different hand postures used, in the wild

H0,d The error rate does not differ between the different hand postures used (see H0,c) in the wild

4.10.4 Subjective Ratings

When analysing subjective ratings, it was seen that gesture input has was been adapted clearly by users who had no experience with it. And the users who had SGK experience did not go back (into STK) and continued using it in their everyday life.

Therefore we have enough evidence to reject the following null hypothesis:

H0,g The user experience of the participants did not differ between STK and SGK in a lab setting

And replace it with this alternate hypothesis:

H1,g Participants had a better user experience with SGK over STK in the wild

4.11 Conclusions

All in all, we could draw the following conclusions from this study:

H1,a SGK has a faster entry rate than STK in the wild

H1,b SGK produces more errors than STK in the wild

H0,c The entry rate does not differ between different hand postures used, when in the wild

H0,d The error rate does not differ between the different hand postures used (see H0,c), when in the wild

H1,e The use of Out of Vocabulary (OOV) words affect SGK more than STK differently in terms of entry rate, in the wild

H0,f The use of Out of Vocabulary (OOV) words do not affect STK and SGK differently in terms of error rate, in the wild

H1,g Participants had a better user experience with SGK over STK in the wild

5

Study C – Comparison of Typing and Speech in a Lab Setting

5.1 Motivation

Even though speech has joined the mainstream text entry methods on a plethora of devices, the HCI research literature offers little empirical evaluation of the current state of affairs in general, and the performance and experience difference between keyboard and speech in particular. Further, most literature focusses on niche aspects of speech such as correcting errors, predicting words better, and was done before speech recognition had its massive improvements in the last few years, due to increased computational power, and breakthroughs in machine learning.

Therefore, the empirical research on how speech vs keyboard state-of-the-art works has been limited in scope, size, and technology form factor, in particular outside a lab environment.

In this chapter, we empirically compare two state-of-the-art text input methods in a controlled lab environment – keyboard and speech. In the next chapter, we empirically investigate how the two input methods perform outside a lab environment.

5.2 Hypotheses

We present the following null hypotheses which are to be accepted or rejectd as a result of this study. As shown, this study is broad and sheds light on many different aspects of text input between keyboards and speech.

H0,a There is no difference in text entry rate between the keyboard and speech in a lab setting

H0,b There is no difference in character error rate after correction between keyboard and speech in a lab setting

H0,c The presence of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of entry rate, in a lab setting

H0,d The presence of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of error rate, in a lab setting

H0,e The user experience of the participants did not differ between keyboard and speech
in a lab setting

H0,f In the lab, users did not prefer to use one method over the other when given a choice between both

H0,g The perplexity of the phrase does not affect keyboard and speech differently in terms of entry rate in a lab setting

H0,h The perplexity of the phrase does not affect keyboard and speech different in terms of error rate in a lab setting

5.3 Variables & Confounds

In this study, we identify three types of variables as independent variables, dependent variables, and confounds.

5.3.1 Independent Variables

These are the variables we explicitly control in this study.

- V1 – Input mechanism (2 levels: Keyboard and Speech)
- V2 – Participant (12 levels: P1-P12)
- V3 – Session (3 levels: S1-S3)
- V4 – Phrase Perplexity (4 levels: PPL1-PP4)

More details on phrase perplexity will be discussed in the Materials section under Phrase Set.

5.3.2 Dependent Variables

These are the variables that we measure as an outcome of this study. The measurements lead to "derived dependent variables" which lead to the analysis of the study results. This means we do not measure these directly but we derive them via calculations from the dependent variables we measure. The following sub sections below describe the variables we measure vs the variables we derive.

5.3.2.1 Measured Dependent Variables

These are the dependent variables we explicitly measure.

5.3.2.1.1 Timestamp at start of entry (T1)

In the case of keyboard, this is the timestamp when the user first begins to type. The time T1 indicates exactly when the first keystroke – or in this case, when the users finger touches the area surrounding the keyboard. On the Android platform, this is normally captured with an *onKeyDown* event.

In the case of speech, this is timestamp when the user presses the SPEAK button, which activates the microphone and speech widget, using the Android Speech Recognition API (Google, 2018d)

5.3.2.1.2 Timestamp when finished entering text (T2)

Theoretically, this is the timestamp when the user enters the last character in the sentence or phrase they intend to type, in the case of keyboard, or when the user stops speaking, in the case of speech.

However, this is impossible to capture programmatically as there is no way a program can know when the user has finished typing – i.e. the key the user just pressed could the last one, or there could be more to come. In speech, this could be captured as when the mic is deactivated (upon a significant spell of silence), however, we do not know if the user wishes to enter more text using speech by reactivating the microphone again.

Therefore when running studies, we use a practical delimiter to capture when the user has completed typing – such as pressing a button which says NEXT, or FINISHED, or performing some other delimiting action, which tells the program that the user has indeed finished typing or speaking. In a texting application this would be denoted by pressing SEND. In this study, we capture the "end of phrase" when the user indicates they want to move to the next sentence by pressing NEXT.

It is obvious that there is a slight delay between entering the last character in the response phrase and pressing next, however, this does not skew the results in the study as:

i. When typing, this happens almost instantaneously

j. When entering text via speech, the user presses NEXT almost instantaneously after getting what they need

k. We explicitly tell the users to use a minimal delay between finishing entering text and pressing next

l. This delay is uniform across the entire study (and does not differ between subjects)

m. If the user does require to proofread what they entered, this should be indeed factored in to the time it takes to enter text using the given input mechanism, as this is a critical factor

5.3.2.2 Derived Dependent Variables

These are the dependent variables we calculate from the measured dependent variables. Descriptions of these can be found in Chapter 1 – Introduction, under Conventions.

- Number of characters in the response phrase (N)
- Typing duration (T)
- The Error (E)
- Entry Rate (WPM)
- Error Rate

5.3.3 Confounding Variables

These are variables that we did not try to control, but still would be consider as variables due to their confounding nature, as they can definitely affect the typing experience and performance in the study. We identify one confounding variable in this study - descriptions of which can be found in Chapter 1 – Introduction.

- OOV words

5.4 Apparatus

5.4.1 Hardware Apparatus

We used a single Huawei P20 Lite mobile devices running Android 8.0 (Oreo). The 5.8" LTPS IPS LCD capacitive touchscreen had a resolution of 1080 x 2280 pixels at 432 pixels per inch. The physical devices measured 148.6 x 71.2 x 7.4 mm.

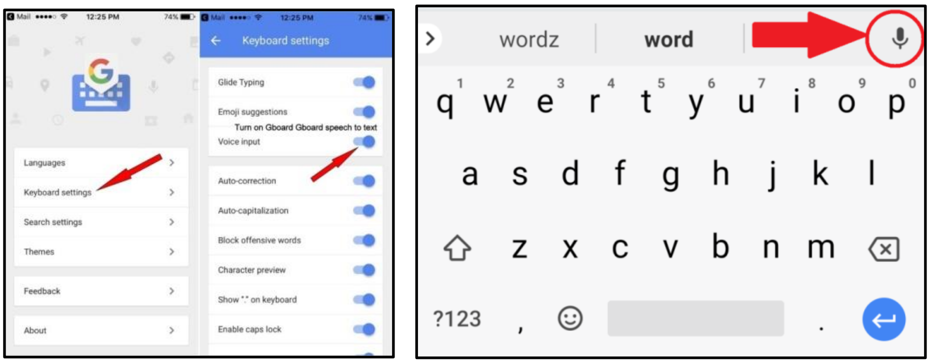
At the time of these studies, the Huawei P20 Lite was a representative option of the Stock Android, which was the main reason behind using this particular brand and model. Full hardware specifications can be found at (GSMArena, 2018)

5.4.2 Software Apparatus

There were two major components in the software apparatus. The First was the Google Keyboard, which had its own implementation of keyboard and speech. The second was the experimental software that was required to run the study.

5.4.2.1 Speech Input on Google Keyboard

Speech input can be invoked on Google Keyboard using the mic button located at the top right of the suggestions bar, as shown in the image below. This will activate the mic and listen for speech input, and once heard; it will infer the text and render it onto the target text box immediately. This feature (if turned off) can be activated by enabling the "voice input key" under Settings for Google Keyboard.



Glide Typing ●
Emoji suggestions ●
Voice input ●
Auto-correction ●
Auto-capitalization ●
Block offensive words ●
Character preview ●
Show "." on keyboard ●
Enable caps lock ●

Figure 46 - Accessing Speech Input on Google Keyboard

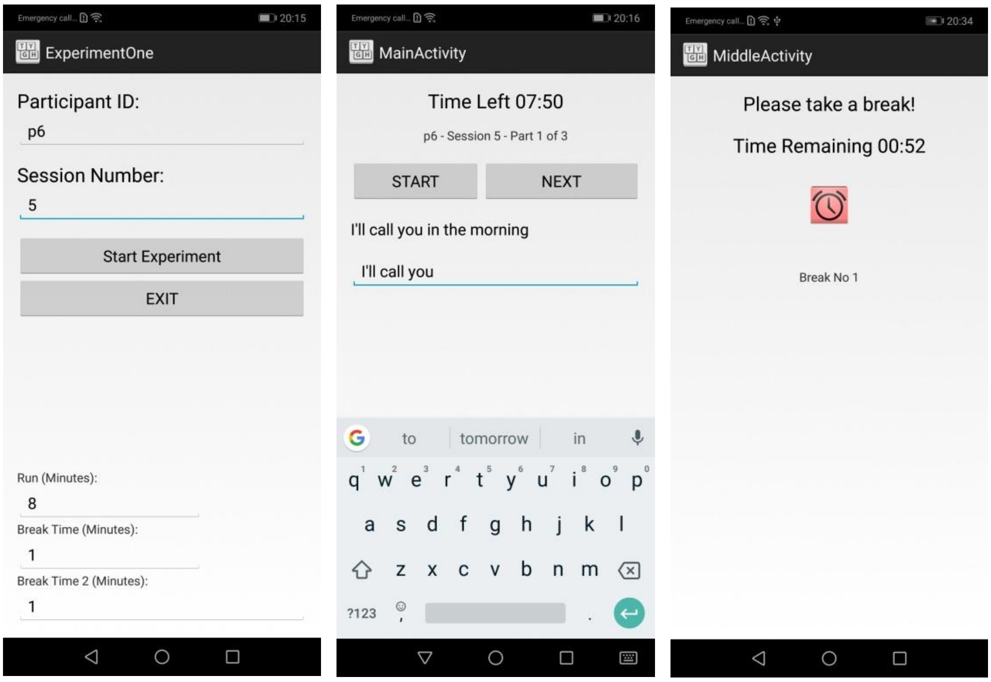
5.4.2.2 Experimental Software

The app used for the study outlined in this chapter is designed for a typical longitudinal study carried out in a lab environment, where a participant would repeatedly enter response phrases to a stimuli phrase shown to them, for a set duration, with breaks in between. The app will record the stimuli and response phrases, and the elapsed time, which could be used to analyse results.

The specialty of this app is that it allows the fully automatic execution of the experiment without the intervention of the researcher. The app assumes two conditions, and allows the researcher to specify which condition to use for each experiment session. The app provides an interface to provide configuration parameters for the experiment. These are a participant identifier, a session identifier, number of continuous typing runs (i.e. 3), the duration of each run (i.e. 8 min), and the break duration between two runs (i.e. 1min). The researcher simply has to provide this information and then hand over the device to the participant, and the participant simply has to press START to begin the experiment. From then onwards, the app will guide the participant through the experiment.

The app will read stimuli phrases from the provided phrase set; will provide a pseudo-randomized copy of it to the participant – see Phrase set under Materials for more details. The phrases come from the famous Enron Mobile Email Dataset (Keith Vertnanen, n.d.) which is also described in Materials section.

As shown in the figure, the participant will be shown an interface where he/she will be required to enter the phrase shown. The countdown timer will keep counting down from the specified run duration value to achieve this). Two timestamps will be captured and written to a file in the background, one corresponding to pressing START, and the second being the time pressed NEXT. Following this, the stimuli and response phrases will also be written to the file. When the user presses NEXT, the app will display the next sentence and clear the textbox. By this time, if the countdown timer has reached zero, pressing NEXT would take him to another activity which would allow the participant to take a break for the aforesaid time This process is then repeated until the all the runs are complete.



Participant ID:
p6

Session Number:
5

Run (Minutes):
8

Break Time (Minutes):
1

Break Time 2 (Minutes):
1

Time Left 07:50

p6 - Session 5 - Part 1 of 3

I'll call you in the morning

I'll call you

Please take a break!

Time Remaining 00:52

Break No 1

Figure 47 - Software Apparatus used for Experiment C

Certain buttons (i.e. Back Key) are disabled by the app to avoid unexpected behaviour that would hinder the experiment. The app requires permission to vibrate and play ringtones, as the user needs to be notified when the break is over and it is time to start entering text again.

5.5 Materials

This section describes the surveys used, the phrase set used for the above study, compensation and phrase set.

5.5.1 Surveys

In addition to capturing the user's performance when entering text using the two mechanisms, we also surveyed their responses on previous typing experience, mobile phone experience, smartphone experience, and perceived performance and user experience, which shall be explained in the upcoming sub sections.

5.5.1.1 Preliminary Survey

This was given at the beginning of the entire study, before the user had the opportunity to enter any text whatsoever. The purpose of the survey was to gauge the prior experience of the user.

Q1. In your life, which text entry method did you use more during your day-to-day activity?

Keyboard 1 2 3 4 5 6 7 Speech

Q2. When using keyboard, which input mechanism do you normally use?

Tapping 1 2 3 4 5 6 7 Gesturing

Rationale (Q1-Q2):

This, in conjunction with the final survey (presented at the end of the study), this will reveal if users had prior experience with speech input, and if not, would have started using speech input in their day to day life as a result of participating in this study.

Q3. What kind of mobile devices do you use? Tick all that apply.

Smartphone – Android
Smartphone – Apple
Smartphone – Microsoft
BlackBerry
Feature phone (no large touchscreen)
Tablet – Android
Tablet – Apple
Tablet – Microsoft
Phablet – Android (A very large smartphone)

Q4. Please write the brands and models of the mobile device you have used the most in last few years.

Brand Model Duration of Use

Q5. Please rate your ability to type using your mobile device.

Very slow typist 1 2 3 4 5 6 7 Very fast typist
Inaccurate typist 1 2 3 4 5 6 7 Very accurate typist
Speech Novice 1 2 3 4 5 6 7 Speech Expert

Rationale (Q3-Q5):

This is to gauge the user’s previous smartphone experience, which is directly attributable to one’s performance on a typing and speech. This will be revisited in the participants section.

5.5.1.1.1 Post-survey

This was given at the end of each session, again to gauge the participant’s hand posture which they used for typing on that particular session (if the condition was typing)

Q6. What posture did you use mostly for this session?

Thumb Single-Finger Two-Thumbs

Q7. Which keyboard did you use for typing?

Tapping 1 2 3 4 5 6 7 Gesturing

Rationale (Q6-Q7):

We wanted to find out which hand postures were used by the participant for this particular session, and which keyboard (tapping or gesturing they used for the typing condition). They were told to be consistent when typing inside each session i.e. not change their hand posture or mechanism, and this was adhered to.

5.5.1.2 Final Survey – End of the study

This was the final questionnaire at the end of the full study. We required the participants to provide qualitative and open ended answers on what they liked and disliked about each input method, and quantitative self-ratings on the perceived performance on each method.

Q8: What did you like about each input method?

Q9: What did you dislike about each input method?

Use your own words and be descriptive as possible. Talk about ease of use, learning curve, speed, accuracy and your user experience (did you feel fatigue after typing/speaking, did you keep getting faster / slower / more accurate / inaccurate etc.)

Q10: Did your everyday text input get affected as a result of this experiment?

(Did you learn a new method of text input (i.e. Speech Input), became aware about a new tool (Google keyboard), apply it to your own day-to-day life, or did you become faster, more accurate etc.) If you would prefer to use either method in real life – i.e. use or not use speech – then why?

Q11: What do you think about this experiment?

Rationale (Q10):

We wanted to find out if this experiment has affected the user's general typing experience in real life. As there were users who were exposed to SGK for the first time, it was possibly the most interesting question in this survey by far. We will revisit this question more in the results section.

Q12: Give us few examples of where you will use keyboard over speech for text entry?

Q13: Give us few examples of where you will use speech over keyboard for text entry?

Rationale (Q12-Q13):

We wanted to find out under which circumstances users would choose speech over keyboard and vice versa, giving us a better understanding on the adaptability of speech as a mainstream text entry mechanism

Q14: How fast do your self-rate your performance on keyboard?

Very slow 1 2 3 4 5 6 7 Very fast

Q15: How fast do your self-rate your performance on speech?

Very slow 1 2 3 4 5 6 7 Very fast

Q16: How accurate do your self-rate your performance on keyboard?

Very slow 1 2 3 4 5 6 7 Very fast

Q17: How accurate do your self-rate your performance on speech?

Very inaccurate 1 2 3 4 5 6 7 Very accurate

Q18: How easy is keyboard to use?

Very inaccurate 1 2 3 4 5 6 7 Very accurate

Q19: How easy is speech to use?

Very easy 1 2 3 4 5 6 7 Very difficult

Q20: How would you rate your preference of one over the other or choose between them

Very easy 1 2 3 4 5 6 7 Very difficult

Rationale (Q15-Q19):

We wanted to find out how the users self-rate their own, perceived, performance on each input method in terms of speed, accuracy and ease of use, regardless of what the performance statistics indicate.

Rationale (Q20):

We wished to gauge how much participants preferred one method over the other

5.5.2 Phrase Set

We used a subset of the Enron mobile email dataset (Keith Vertanen & Kristensson, 2011) with the following conditions:

- Each sentence should be less than 60 characters in length
- No numbers
- No special symbols
- Sentences with varying perplexity from 1.68941 - 13.518

This resulted in phrase set of 1016 phrases. We counted 1382 unique words in this test set. The rationale behind this was we didn't want users to switch between keyboards to enter numbers and special symbols – i.e. in Android, when using Google Keyboard; users have to change the view back and forth to enter numbers, symbols and letters.

Further, there is no guaranteed way in speech recognition to enter punctuation marks or numbers. I.e. if the user speaks "two" the speech engine might infer the word "two" or the number "2", similarly with punctuation i.e. if the user speaks "question mark" it may result in the word or the symbol. Therefore we decided these to be explored in a different study.

5.5.2.1 OOV Words

We did not, however, exclude sentences with Out of Vocabulary words (OOVs). We did this because we want to find out how each input mechanism performed differently when OOV words were part of the mix – especially speech. As speech has no way of inferring OOV words, it was an interesting observation as to how it affected the typing speed, the error rate, the user experience and most of all how users dealt with this particular issue when typing.

We compared all the words in the phrase set against a standard lexicon (64K common words used in the English language). The words that weren't in the lexicon were each entered carefully on the Google keyboard, by tapping the center of each key on the STK, by gesturing from the center to center of each key on the SGK, and clearly pronouncing the word into the voice input of Google keyboard multiple times. We noted that the same 39 words were out of vocabulary (OOV) words for STK, SGK, and speech. These OOVs appeared in 40 sentences (3.94% of 1,016) in the phrase set, and were marked as sentences with OOV words. These OOV sentences were analysed in post-hoc analyses after the experiment.

5.5.2.2 Non OOV Samples

The following are a few sample phrases from the set which do not include OOV words

'Any time Thursday'
'Can we have them until we move'
'I compliment you'
'Jan has a lot of detail'
'Still waiting on decision'

5.5.2.3 OOV Samples

The following are a few sample phrases from the set which include OOV words

'Have we assigned employees to NetCo'
'He will walk Tanya Rohauer through the exact same steps tomorrow'
'If so Whitt is done'

'It should be Cynthia Barow instead'

'John Keffer is the one I know best'

5.5.2.4 Sentence Perplexity

We wanted to find out if sentence perplexity affects speech and keyboard at the same level, therefore we calculated the perplexity of each phrase in the chosen set and divided them into 4 quartiles as follows.

Quartile	Min PPL	Max PPL	Example Sentences
Q1	1.68941	2.33612	I plan to be in the office tomorrow (2.13943)
Q2	2.33715	2.68698	Are you in a good mood (2.48629)
Q3	2.68758	3.14593	This is the crew (2.90663)
Q4	3.14939	13.518	He will walk Tanya Rohauer through the exact same steps tomorrow (3.62079)

5.5.2.4.1 Calculating Sentence Perplexity

We calculated the perplexity of these 1016 Enron mobile sentences under a 12-gram language model trained on billions of words of mobile-like data. We limited to sentences with between 3 and 12 words. The average Turk worker memorization CER had to be between 0 and 10%. We excluded sentences with punctuation other than apostrophe and end of sentence punctuation, and sentences with the digits 0-9. We also removed sentences with words of 2+ letters in capitals (usually abbreviations). The ARPA format gzipped language model is 1012MB in size (it is only lightly pruned). The vocabulary size is 34: A-Z plus .,'! and pseudo-word for space.

Credits go to my co-author on the manuscript "Performance and User Experience of Typing and Speech on Mobile Devices in a Lab Setting and in the Wild", Keith Vertanen for the preparation of this phrase set, which he developed for his own research for similar purposes.

5.5.2.5 Ordering of Phrases

When the phrases were displayed to the participants during the study, they were chosen at pseudo-random – every 4 sentences shown to the participant were chosen at random from each of the Quartiles above, ensuring that each participant on each session entered an equal number of sentences from each Quartile.

The Fisher Yates shuffling Algorithm (Fisher & Yates, 1948) was used inside each quartile to randomize each subset, and then was simply merged together using a “merge in turn” algorithm. The two algorithms can be outlined as follows.

```
// ar is the array that needs shuffling

public static void fisherYatesShuffle(int[] ar) {
    // generate a randomizer
    Random rnd = ThreadLocalRandom.current();

    for (int i = ar.length - 1; i > 0; i--) {
      // generate random index between 0 and (i+1)
      int index = rnd.nextInt(i + 1);

      // perform a simple swap between the current and
      // random positions
      int a = ar[index];
      ar[index] = ar[i];
      ar[i] = a;
    }
}
```

```
// ar1, ar2, ar3, ar4 are 4 equally sized arrays
// each representing one randomized quartile of the phrase set

public static int[] mergeInTurn(int[] ar1, ar2, ar3, ar4) {
    int[] result = new int[ar1.length * 4];
    int index = 0;

    for (int i=0; i<ar1.length; i++) {
        result[index++] = ar1[i];
        result[index++] = ar2[i];
        result[index++] = ar3[i];
        result[index++] = ar4[i];
    }

    return result;
}
```

5.6 Compensation

Participants were compensated £20 for their time in amazon vouchers. The standard rate of compensation in University of St Andrews is £5 per hour for participating in experiments. Given that each participant had to attend 6 sessions of roughly 40 minutes each, their commitment was 4 hours.

Further we offered an incentive of an extra £10 for the fastest performing participant in each condition, under a certain error rate threshold.

5.7 Participants

We recruited 12 volunteers from the University of St Andrews campus, details of whom are described as follows. As per the previous studies, the participant number was justified by previous studies performed in literature (P. O. Kristensson & Denby, 2009)

5.7.1 Participant Demographics

Due to the ethics agreement we cannot publish any identifiable information about the participants - therefore the aggregate results of each demographic will be described below and not attributed to individual participants.

5.7.1.1 Gender

We had 9 female participants and 3 male participants.

5.7.1.2 Age

The age range of the participants was 17-35, with average of 23.4 and standard deviation of 5.1. This ensured we had a satisfactory distribution of ages which is quite representative of the real world population who uses smartphones for text entry in the year 2018 (when this experiment was performed).

5.7.1.3 English Proficiency

Four participants were native English speakers, and the rest used English as their second language. Given they were all doing either an undergraduate, postgraduate or PhD in University of St Andrews they had to be proficient in English if not they would not be admitted for study – as per the English language requirements of university admissions - getting a 7.0 or above in IELTS (“IELTS,” n.d.).. This ensured that our participants were able to understand, read and copy the sentences in the above phrase set without difficulty.

5.7.1.4 Geographic Distribution

The best part about running studies in University of St Andrews is that it attracts students from all over the world. In a recent survey, it was found that St Andrews students represent 120 different cultures, which gives a mini sample of the global population. Our study therefore, had participants from 10 different countries. The 12 participants were distributed across the globe as follows.

Poland	USA	Sri Lanka	Pakistan	India	Scotland	Slovakia	England
1	2	2	1	3	1	1	1

5.7.1.5 Accent Distribution

Since this study involves speech recognition, it is important to gauge the accents of the participants, when speaking English. We asked the participants to provide a self-assessment of their own accent/dialect when speaking English for this study and the results were as follows.

Eastern Europe	Chicago	New York	Other Asian	Indian	Glasgow	South London
2	1	1	2	4	1	1

5.7.1.6 Field of Study Distribution

The participants field of study also varied across the disciplines – this also ensures that our study was rich in terms of different levels of technical expertise and not include participants from either a daily high tech usage demographic (e.g. Computer Science) or low (e.g. the Humanities). The participant’s fields of study can be summarised as follows.

Computer Science	Chemistry	Languages	Psychology	Geography	International Relations	Economics
6	1	1	1	1	1	1

Further, the participants’ level of study can be summarised as follows.

Undergraduate 1st Year	Undergraduate 2nd Year	Undergraduate 4th Year	Masters	PhD
2	1	1	5	3

5.7.1.7 Smartphone Experience

By interviewing the participants, we gauged their previous smartphone experience and found the following. Three participants were Android users, and 9 of them were iPhone users.

5.7.1.8 Exposure to STK & SGK Keyboards & Speech

Again by interviewing the participants, we gauged their previous experience with STK and SGK, and speech, which was very important for this study. All the users were experienced with QWERTY keyboard

Experience with QWERTY	Experience with STK	Experience with SGK
all (three android users) (9 iPhone users)	all (three android users) (9 iPhone users)	3 (three android users)

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
Exposure to SGK (1 – low, 7 – high)	1	6	1	1	1	1	1	1	4	1	1	1
Speech Input Competence (1 – low, 7 – high)	3	1	3	1	3	4	1	6	3	4	1	3
Perceived speed when typing	5	5	4	5	6	5	6	6	4	5	5	4
Perceived accuracy when typing	5	5	4	5	6	6	6	6	5	6	2	4

The participants provided the following values on a 7 point Likert scale for their previous experience with speech input on their mobile device. The Android users have been highlighted.

5.7.2 Design

We incorporated a within-subjects design for this study - meaning each participant had to experience both conditions (typing and speech). The participants were exposed to alternating conditions of typing and speech. The participants were allocated randomly to the two groups.

5.7.2.1 Screening

Participants were screened for:

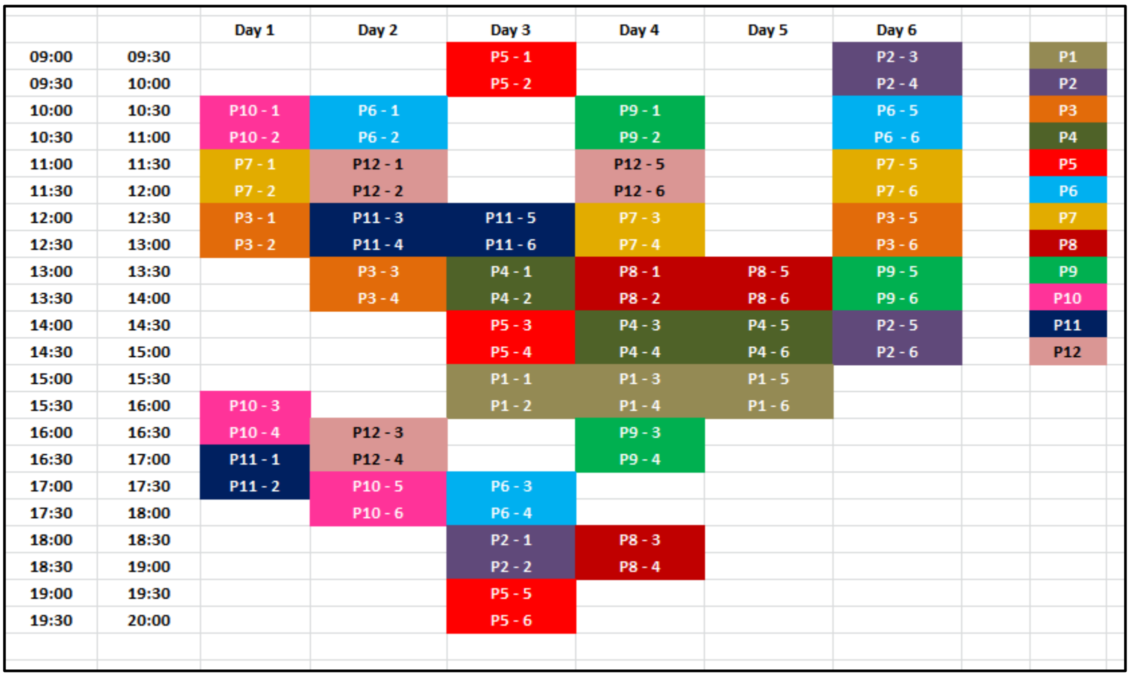
(g) English proficiency – either they had to be native English speakers or use English as a second language in their day to day life / studies. They had to be able to write and speak English proficiently.

(h) Experience with QWERTY

(i) Experience with smartphones

5.7.3 Scheduling

The scheduling of participants was done in the manner below. The participants had to attend for 6 sessions in total, grouped into pairs –one session for each condition (typing and speech). Each session-pair was spaced at least 4 hours apart and at most 3 days.



		Day 1	Day 2	Day 3	Day 4	Day 5	Day 6		
09:00	09:30			P5 - 1			P2 - 3		P1
09:30	10:00			P5 - 2			P2 - 4		P2
10:00	10:30	P10 - 1	P6 - 1		P9 - 1		P6 - 5		P3
10:30	11:00	P10 - 2	P6 - 2		P9 - 2		P6 - 6		P4
11:00	11:30	P7 - 1	P12 - 1		P12 - 5		P7 - 5		P5
11:30	12:00	P7 - 2	P12 - 2		P12 - 6		P7 - 6		P6
12:00	12:30	P3 - 1	P11 - 3	P11 - 5	P7 - 3		P3 - 5		P7
12:30	13:00	P3 - 2	P11 - 4	P11 - 6	P7 - 4		P3 - 6		P8
13:00	13:30		P3 - 3	P4 - 1	P8 - 1	P8 - 5	P9 - 5		P9
13:30	14:00		P3 - 4	P4 - 2	P8 - 2	P8 - 6	P9 - 6		P10
14:00	14:30			P5 - 3	P4 - 3	P4 - 5	P2 - 5		P11
14:30	15:00			P5 - 4	P4 - 4	P4 - 6	P2 - 6		P12
15:00	15:30			P1 - 1	P1 - 3	P1 - 5			
15:30	16:00	P10 - 3		P1 - 2	P1 - 4	P1 - 6			
16:00	16:30	P10 - 4	P12 - 3		P9 - 3				
16:30	17:00	P11 - 1	P12 - 4		P9 - 4				
17:00	17:30	P11 - 2	P10 - 5	P6 - 3					
17:30	18:00		P10 - 6	P6 - 4					
18:00	18:30			P2 - 1	P8 - 3				
18:30	19:00			P2 - 2	P8 - 4				
19:00	19:30			P5 - 5					
19:30	20:00			P5 - 6					

Figure 48 - Participant Schedule for Experiment C

5.8 Procedure

The execution of the study was done in two steps. The first was a pilot where the authors of the paper used the apparatus to find any errors in the implementation that could affect the study, followed by the actual study involving the 12 participants. The pilot study did not yield any problems therefore the actual study was commenced without further problems.

As explained above, the experiment consisted of six sessions split into three sessions for typing and three sessions for speech. The sessions were grouped into pairs, one for keyboard and one for speech and scheduled consecutively. We divided the participants into two equal groups. Participants in the first group encountered speech first, the other group encountered keyboard first. The session pairs were spaced at least four hours apart and were maximally separated by three days. Each session consisted of three 8-minute-long runs followed by one-minute-long breaks.

The experiment used a transcription task where participants were shown a phrase from the dataset and asked to copy it. They could use either keyboard or speech to copy the task depending on the condition they were in.

We encouraged participants to focus on both speed and accuracy by providing an additional £15 Amazon voucher as an incentive to the fastest and the most accurate participants. Whilst being encouraged to use the Google keyboard's suggested words for correction, we discouraged participants to go back and correct errors unless absolutely necessary. I.e. when transcribing the phrase, if an error or typo was made, we asked the participants to decide if they would send this message off if a human user was listening on the other end. If they decided the original word was still able to be deciphered by the recipient, then they were not required to go back and correct it. When using speech, the participants were asked to continue or re-correct the sentence based on a simple decision making process – assuming a human being is receiving the message on the other side, if he/she is able to get the message without much difficulty, then they were asked to continue. However, if the entered sentence was significantly off the ballpark from the stimulus sentence, then they were asked to go back and correct it.

Participants were seated during the experiment, with no distractions from the environment. Our experiment app recorded the stimulus phrases and the response text using millisecond timestamps when the user pressed START and when the user pressed NEXT.

Participants rated their previous experience with software keyboards (STK and SGK) and speech on mobile devices, and self-rated themselves on how fast and accurate they thought they were.

We intentionally did not control hand posture. Instead we asked participants to use their preferred posture and report it at the end of each session. It was interesting to see that most users were iPhone users and were only familiar with STK typing. Another interesting observation was that all the users used two-thumb touch-typing during all the sessions that involved typing. At the end, participants were asked to write descriptive and open comments about what they liked and/or disliked about each text entry method.

5.9 Results & Analysis

In total we collected ~15 hours of typing data – 8 minutes of typing per run, 3 runs per session, 3 sessions per participant x 12 participants. These 15 hours of typing contained 4,395 data points, of which 82 we discarded as outliers – as being 3 standard deviations away from the mean. The remaining 4,313 data points have been used in the subsequent analysis.

We also collected ~15 hours of speech data – 8 minutes of speech input per run, 3 runs per session, and 3 sessions per participant x 12 participants. These 15 hours of speech input contained 6,381 data points, of which 64 we discarded as outliers – as being 3 standard deviations away from the mean. The remaining 6.317 data points have been used in the subsequent analysis.

5.9.1 Data Preparation

For the purpose of analysis, the data was divided into 9 blocks. Each block represents a 8 minute long typing run (3 such blocks in a session, and 3 sessions per condition per participant)

The following scatter plot visualised the data points in the typing condition.

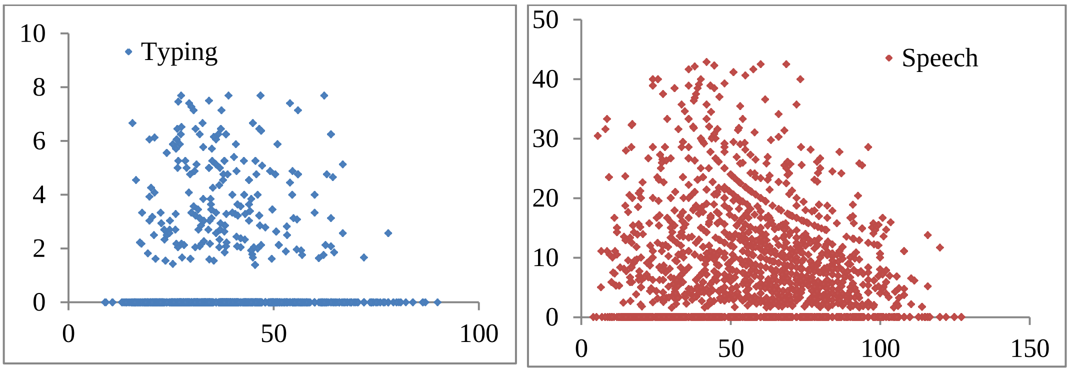


Figure 49 - Entry Rate vs Error Rate Scatter Plots for Experiment C

The following scatter plot visualises the data points in the speech condition.

5.9.2 Entry Rate

As per the previous studies, our major interest was the entry rate comparison between the two studies. For the purpose of analysis as mentioned above, each typing run is represented as a block. It was interesting to see that in the lab, the speech condition was on average much faster than the typing condition.

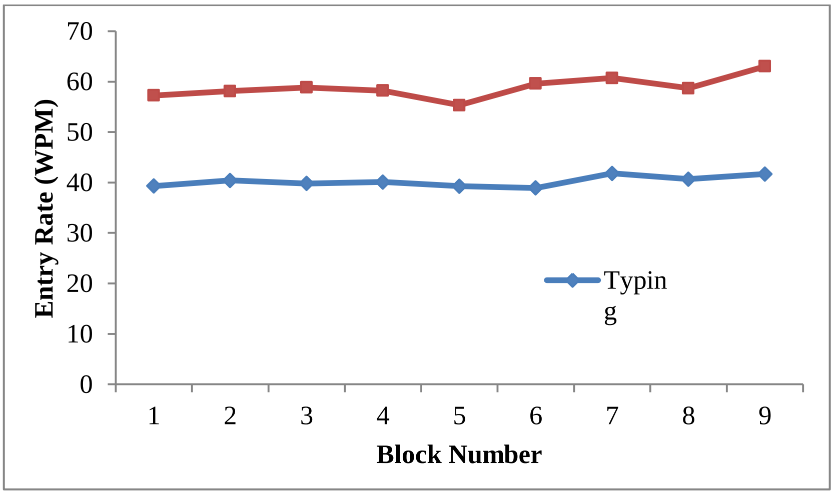


Figure 50 - Entry Rate vs Block for Typing and Speech in Experiment C

We conjecture that this is because users are able to speak much faster than they type – ~200 word per minute vs ~65 words per minute – and the state of the art speech recognizers have started to live up to this expectation. However the rate is not high as 200 WPM is due to the fact that it takes a short (yet substantial) duration for the speech recogniser to kick in, recognize the phrase and output the resultant text – which should of course be and is factored into the entry rate.

5.9.2.1 Tests of Within-Subjects Effects

This tells us that speech was significantly faster than typing, participants improved with time, and the interaction between input method and block was not significant.

WPM	F_{df}	df	η_p^2	p
Input Method	132.350	1	.923	**.000**
Block	8.287	8	.430	**.000**
Input Method x Block	1.368	8	.111	.222

5.9.3 Character Error Rate

This had the reverse performance of the entry rate results. Speech recognition had a much higher character error rate then typing.

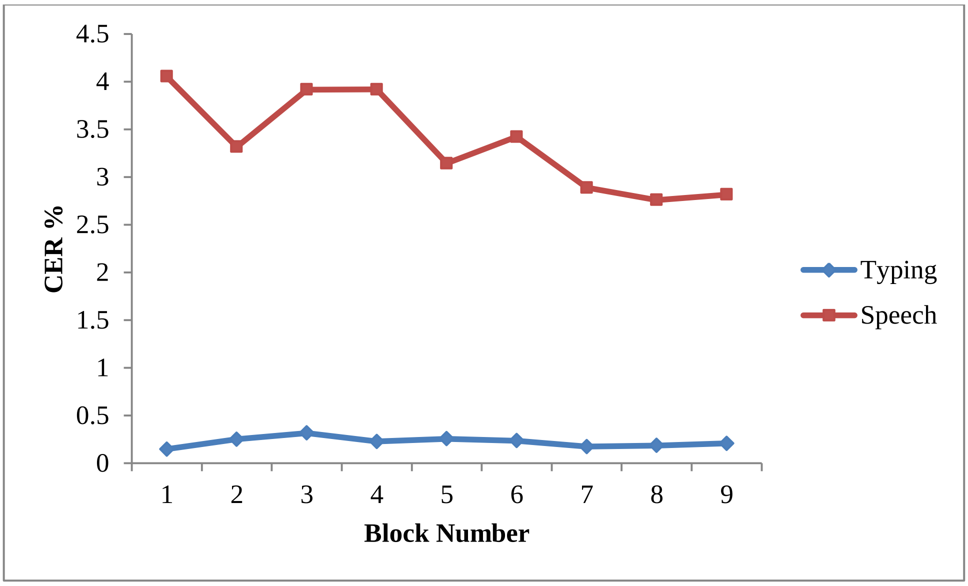


Figure 51 - CER vs Block for Typing and Speech in Experiment C

We claim this to be is the upper bound performance for state of the art speech recognizers vs keyboard in current mobile devices – as the users were fully focussed, seated in a non-distracting, quiet environment, with full visual feedback. The users used the fastest text input mechanism in lab environment – the STK (see Chapter 3), with the two-thumb hand posture. For speech, there was no background noise and super high speed WIFI connectivity at the time of the study which was not shared between any other apps (no other apps were open on the experimenters phone) so the Android Speech Recognizer had no problems in decoding the speech input at the optimal level possible.

5.9.3.1 Tests of Within-Subjects Effects

This tells us that speech had significantly more errors than typing, and the error rate dropped with time, and the interaction between input method and block was significant.

CER%	F_{df}	df	η_p^2	p
Input Method	50.252	1	.820	**.000**
Block	3.280	8	.230	**.003**
Input Method x Block	2.993	8	.214	**.005**

5.9.4 Word Error Rate

The Word Error rate follows a close pattern to the Character Error rate, but with higher values for both typing and speech conditions.

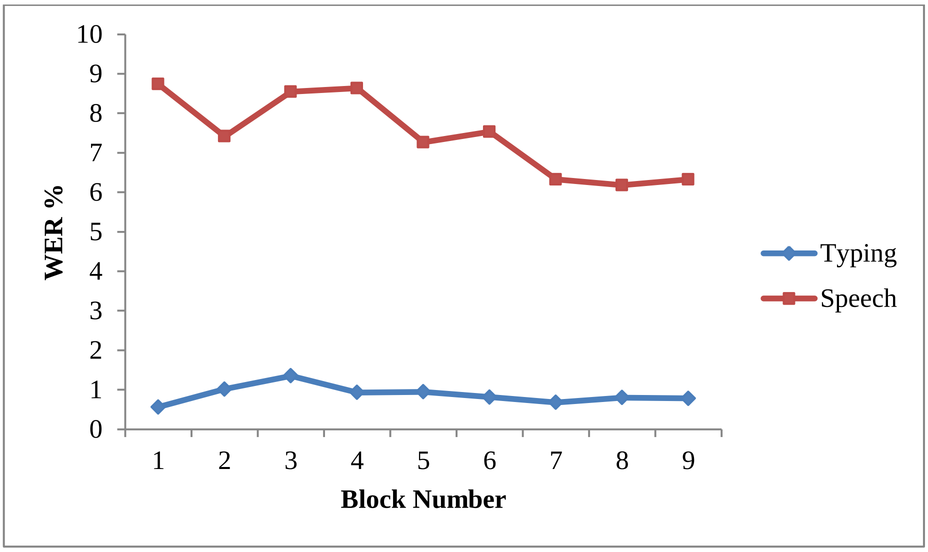


Figure 52 - WER vs Block for Typing and Speech in Experiment C

5.9.5 Evaluating phrases without OOV words

As per the previous two studies, we were interested in identifying what happens when OOV words were removed from the study.

5.9.5.1 Entry Rate

It was seen that the entry rate raised slightly in both typing and speech conditions, but didn't make a difference to the overall result – speech was still faster.

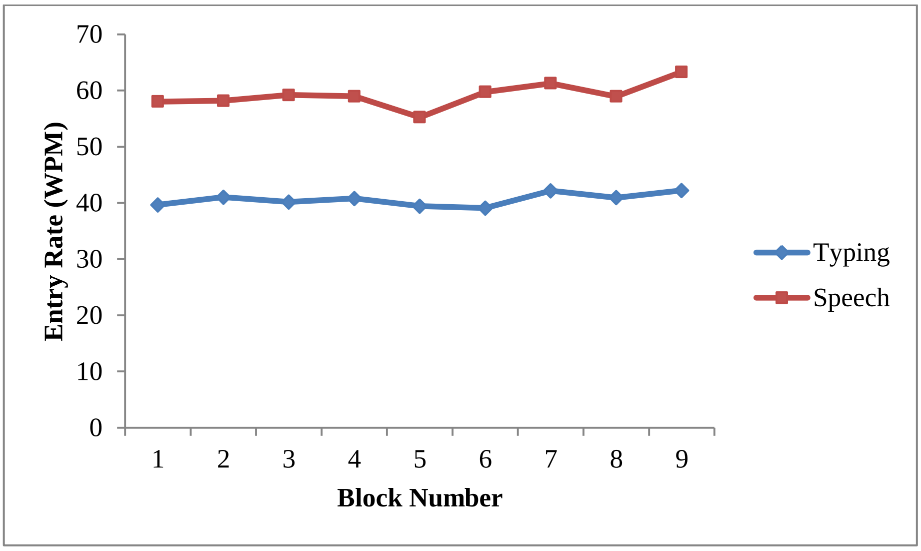


Figure 53 - Entry Rate vs Block for Typing and Speech in Experiment C – excluding OOV words

Tests of within subject's effects say:

WPM	F_{df}	df	η_p^2	p
Input Method	131.572	1	.923	**.000**
Block	8	8.357	.432	**.000**
Input Method x Block	1.360	8	.225	.110

There was a significant effect of input method – i.e. speech was significantly faster than typing, there was a significant effect of block – participants improved over the sessions, and there was no significant interaction between input method and block.

5.9.5.2 Error Rate

The error rate followed a similar pattern with the previous comparison for both CER and WER. It can be noticed that the error rate for speech drops quite sharply toward the end of the study. The following plots and table show the trends and test of within subject's effects.

CER%	F_{df}	df	η_p^2	p
Input Method	47.952	1	.813	**.000**
Block	1.900	8	.147	.070
Input Method x Block	1.578	8	.125	.143

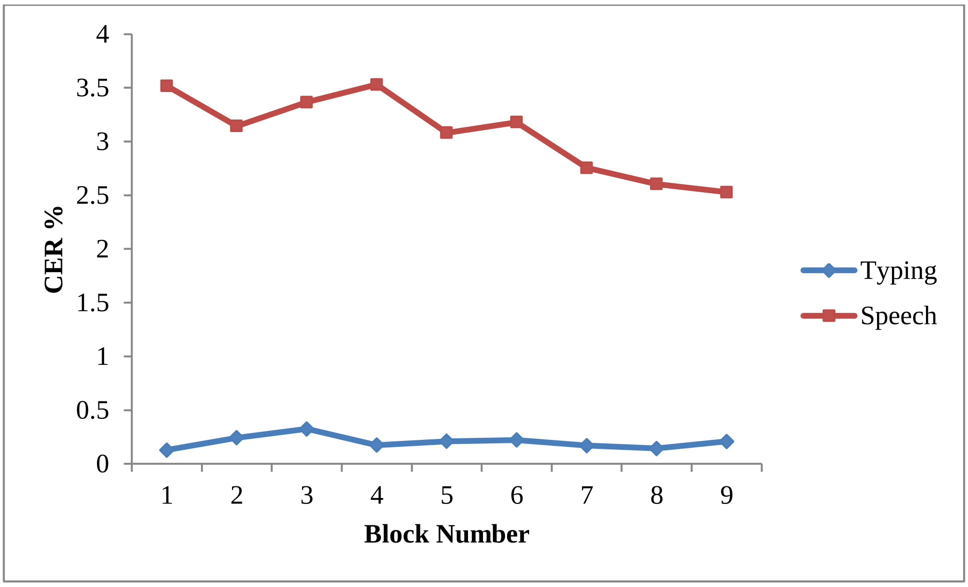


Figure 54 - CER vs Block for Typing and Speech in Experiment C – excluding OOV words

Repeated measures of variance say that speech produced significantly more errors than typing; however the error rate did not significantly differ across the sessions. Also the interaction between input method x block was not significant.

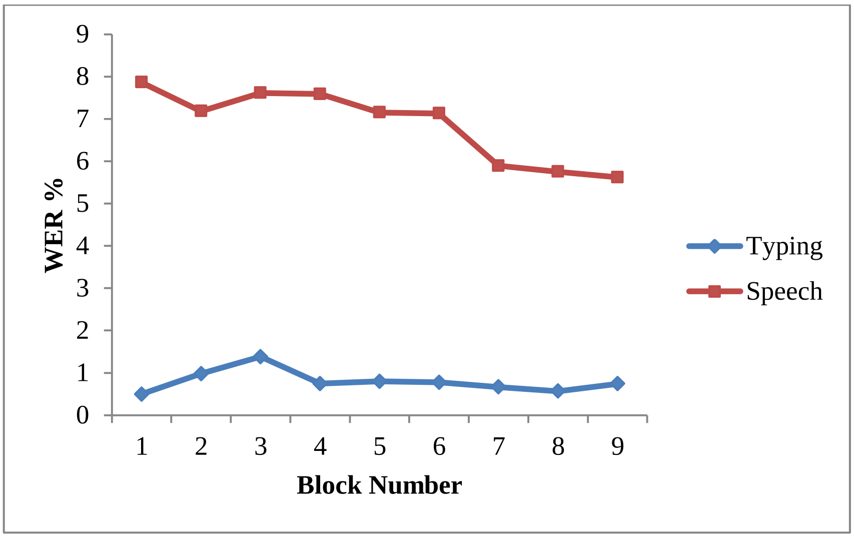


Figure 55 - WER vs Block for Typing and Speech in Experiment C – excluding OOV words

5.9.6 Phrases with OOV sentences

We were interested in the opposite as well. When considering sentences with contained at least one OOV word in it, we found the following results.

5.9.6.1 Entry Rate

The entry rate for typing and speech both dropped sharply – around 10WPM, however, the effect on speech was slightly more than for typing. We believe this is due to the fact that users finding it difficult to immediately go back and correct text on speech as quickly as keyboard.

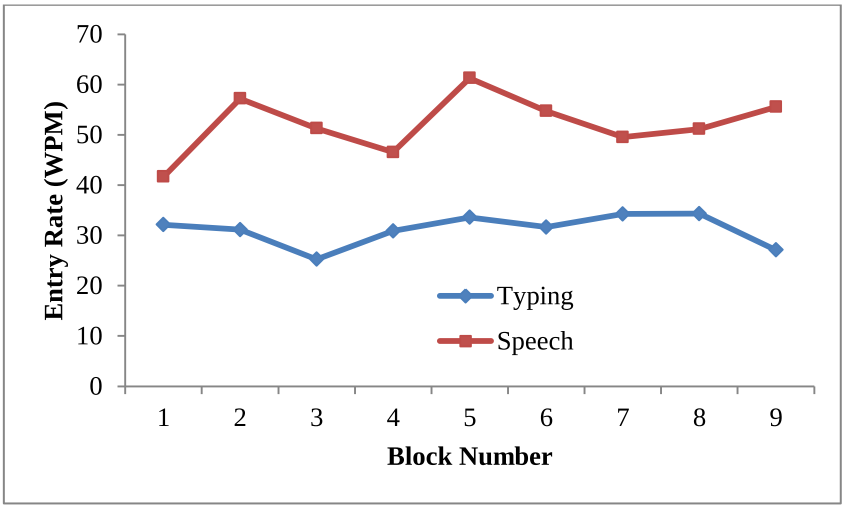


Figure 56 - Entry Rate vs Block for Typing and Speech in Experiment C – considering only OOV sentences

Tests of within subjects effects:

WPM	F_{df}	df	η_p^2	p
Input Method	49.738	1	.819	**.000**
Block	2.892	8	.208	**.007**
Input Method x Block	2.743	8	.200	**.009**

The entry rates are significantly different, speech producing a much higher entry rate than typing. There is also a significant effect of block, the entry rate changes significantly across the sessions. And the interaction between block x input method is also significant.

5.9.6.2 Error Rate

The character error rate (CER) and word error rate (WER) follow a complex pattern – yet clear enough that OOV words affected speech a lot more than typing.

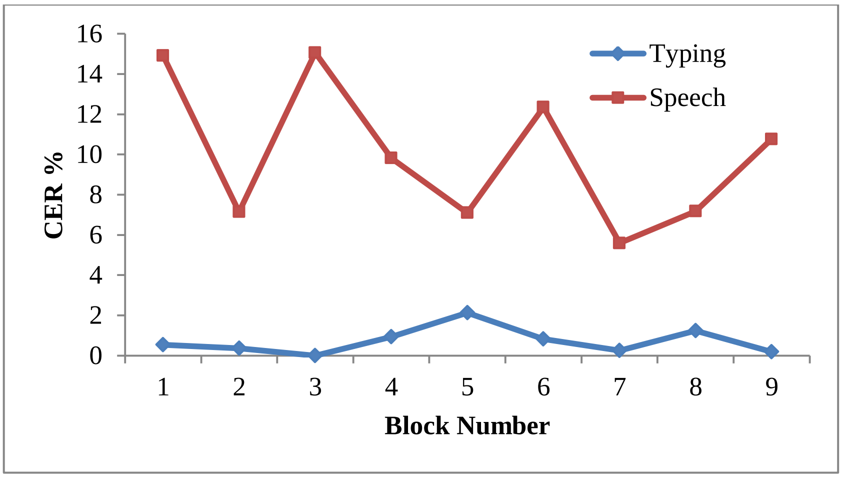


Figure 57 - CER vs Block for Typing and Speech in Experiment C – considering only OOV sentences

Tests of within subjects effects:

CER%	F_{df}	df	η_p^2	p
Input Method	38.980	1	.780	**.000**
Block	2.923	8	.210	**.006**
Input Method x Block	2.903	8	.209	**.006**

This states that speech has a significantly higher error rate than typing. The error rate also significantly varies across the blocks and the interaction between block x input method is also significant.

The word error rate went high as 26% when considering this particular set of phrases with at least one OOV word in them.

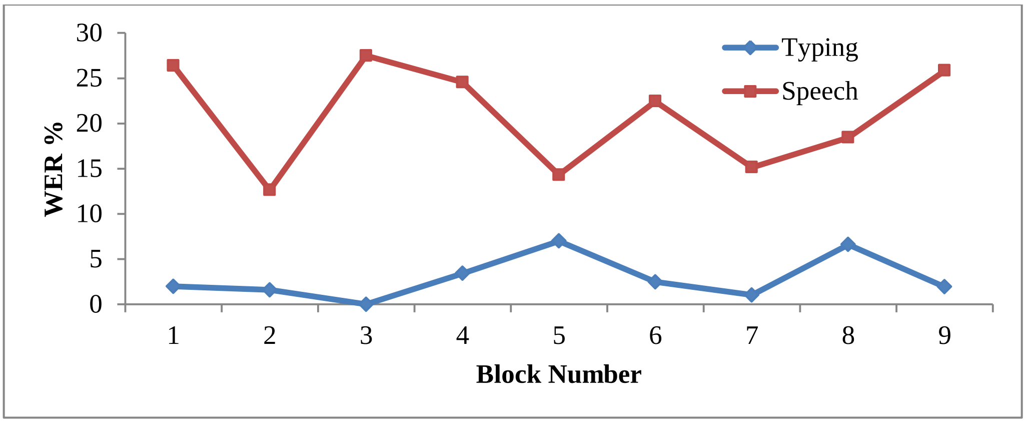


Figure 58 - WER vs Block for Typing and Speech in Experiment C – considering only OOV sentences

5.9.7 Perplexity Analysis

As explained in the Materials section under Phrase Set, we divided the phrase set into quartiles based on perplexity. Based on this, we analysed how the entry rates and error rates would differ between typing and speech with varying perplexity.

5.9.7.1 Entry Rate

When perplexity increases, the entry rate dropped slightly for both the input mechanisms, yet speech was still faster overall. However, the varying perplexity had more of an effect on speech than keyboard.

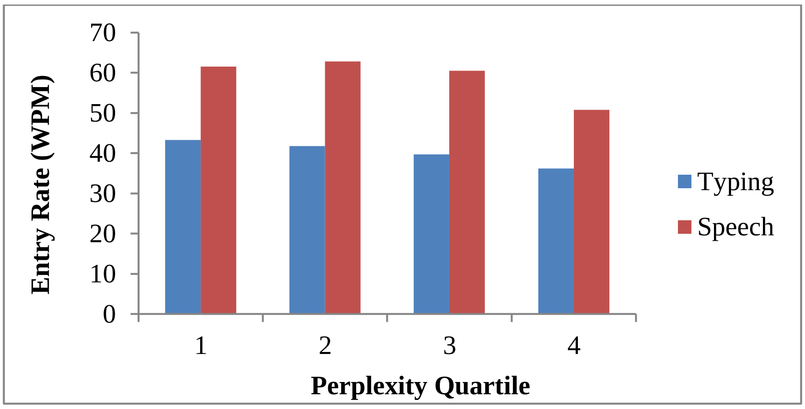


Figure 59 - Entry Rate vs Perplexity for Typing and Speech in Experiment C

The descriptive statistics for the entry rate are as follows

WPM	Typing		Speech	
	Mean	SD	Mean	SD
Quartile 1	41.7017	8.19585	60.4050	7.01526
Quartile 2	39.8508	8.89878	61.3417	8.31946
Quartile 3	38.0025	7.98725	58.5967	9.94705
Quartile 4	34.8492	7.37086	48.6017	10.36324

Test of within subjects effects are as follows:

WPM	F_{df}	df	η_p^2	p
Input Method	141.078	1	.928	**.000**
Quartile	91.553	3	.893	**.000**
IM x Q	17.410	3	.613	**.002**

The interaction IM x Q clearly states that speech is affected more than typing by perplexity.

5.9.7.2 Error Rates

This is possibly the most interesting result of all, which sheds light on the ability of speech recognition to deal with complex sentences. When the perplexity increased, the error rate (both CER and WER) raised geometrically for speech, whereas the error rate for typing stood somewhat steady.

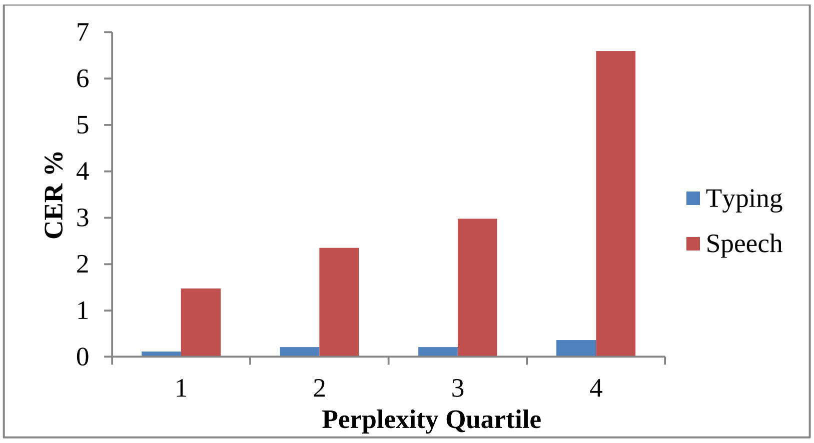


Figure 60 - CER vs Perplexity for Typing and Speech in Experiment C

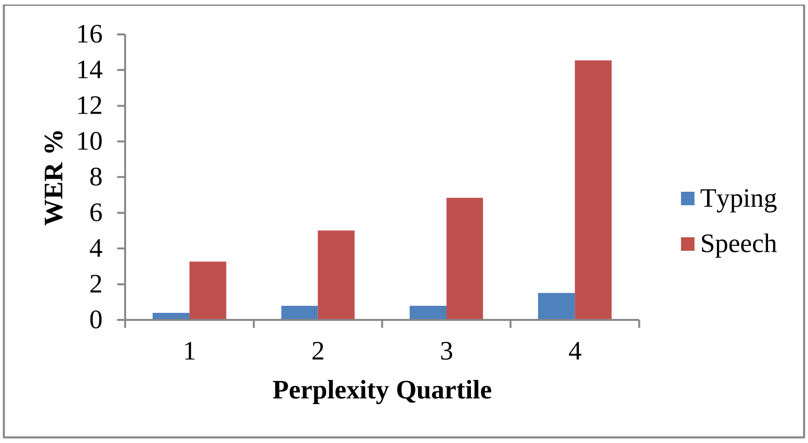


Figure 61 - WER vs Perplexity for Typing and Speech in Experiment C

Tests of within subject's effects says:

CER%	F_{df}	df	η_p^2	p
Input Method	46.543	1	.809	**.000**
Quartile	26.023	3	.703	**.000**
IM x Q	35.094	3	.761	**.000**

Noting the significant interaction between Input Method x Quartile - this clearly states that speech was affected more by perplexity than typing during this study.

The descriptive statistics for the above CER% are as follows:

CER%	Typing		Speech	
	Mean	SD	Mean	SD
Quartile 1	1.6600	1.07486	.4450	.35671
Quartile 2	2.9525	2.22555	.5542	.38429
Quartile 3	3.6058	2.20086	.4233	.41174
Quartile 4	8.6508	3.82146	.6908	.46418

5.9.8 Subjective Ratings

This is evidence that users subjective ratings did not differ significantly between keyboard and speech in terms of perceived speed, accuracy, ease of use and preference/choice.

Median Ratings	Perceived Input Speed (1-7 Likert)	Perceived Accuracy (1-7 Likert)	Ease of Use (1-7 Likert)	Choice (1-7 Likert)
Typing	5.5	5.0	6.0	2.5
Speech	6.0	5.0	6.0	5.5

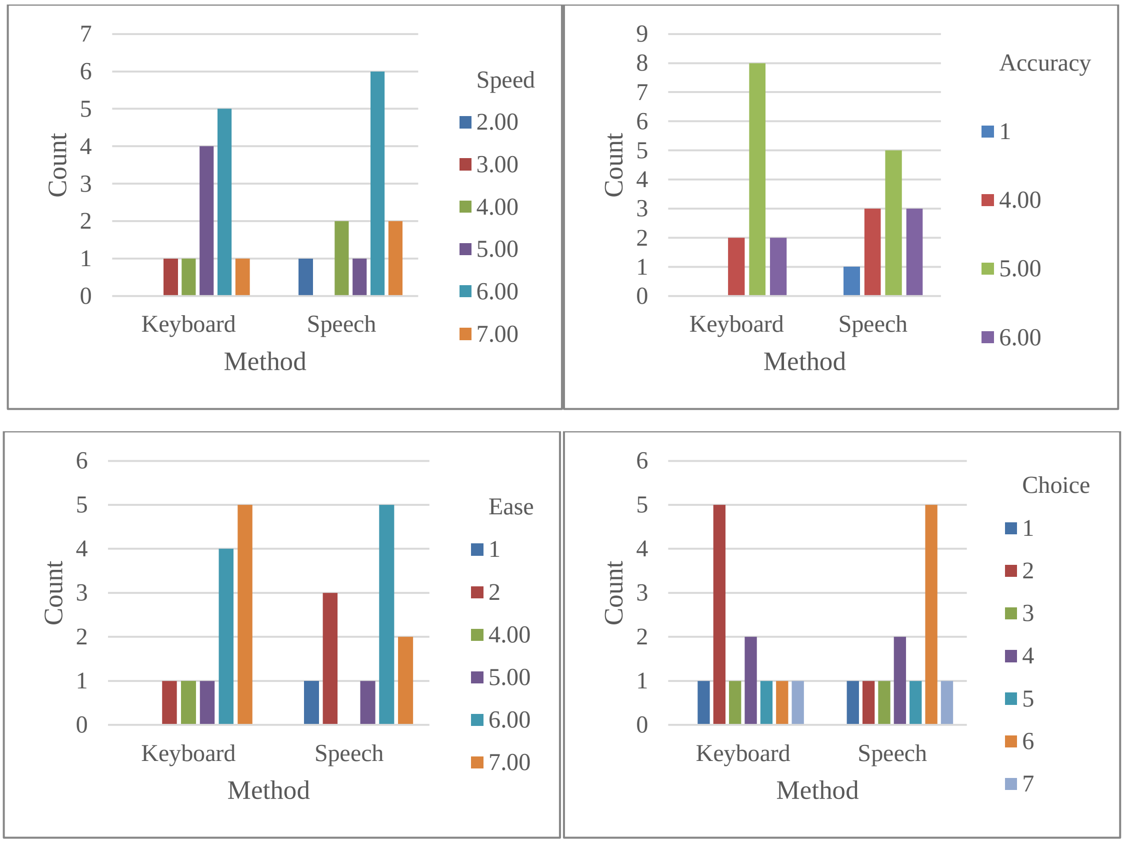


Figure 62 – Likert Scale Ratings for Choice Accuracy Ease and Speed in Experiment C

CHI Squared Tests performed on each variable yields the following:

	Value	**df**	**Asymptotic Significance (2-sided)**
Speed	4.558[a]	5	0.472
Accuracy	2.092	3	0.553
Ease	4.397[a]	5	0.494
Choice	5.333[a]	6	0.502

5.9.9 Open Comments

We also asked the users to provide open comments on what they liked and disliked about each input method, and what would be their decision making factors when choosing one over the other.

5.9.9.1 Like about speech

1. "Much faster than typing"
2. "For short sentences accuracy is very good"
3. "Very easy to use and no learning curve"
4. "Good recognition most of the time"
5. "Don't need to speak robotically. Natural speech is understood"
6. "It wasn't tiring"
7. "It recognised my natural accent"
8. "quick; was surprised it understood me quite well even though I have a strong accent"
9. "Speech method was good because it had to no use of fingers and hence it required less effort and was more easy to use"
10. "I had never used speech before, but found that it was easy to pick up I was impressed with how fast it inputted the words straight after I said them too"
11. "Better than I thought; surprising accuracy given accent"
12. "It's actually faster than typing; more convenient e.g.: while driving you can still interact with the device and safe time also sometimes the weather isn't good so typing is difficult (cold winters when wearing gloves , then speech method comes handy)"
13. "Faster than using keyboard certainly; It was fairly easy to get used to using speech instead of keyboard"
14. "All I had to do was speak instead of move my thumbs"

5.9.9.2 Dislike about speech

15. "When there's background noise or when travelling in public transport it is not convenient to use"
16. "When saying long complex sentences especially with names the accuracy is not good"

17. “Unable to add punctuation and numbers”
18. “It is unable to identify accents of each individual and accent variations of the same person”
19. “Very difficult to enter abnormal words or names or places”
20. “Sometimes predicts correctly and goes back and changed into something incorrect”
21. “Going back editing and pointing the cursor at the right spot is tedious”
22. “Sometimes you have to correct the result using a keyboard”
23. “Other people van hear your conversation”
24. “No speech navigation or delete options”
25. “Sometimes it completely missed the words and transcribed something massively irrelevant”
26. “Some accents aren't properly recognised”
27. “Proper nouns were hard to translate into the correct spelling. I had to articulate slowly or else the words wouldn't pick up”
28. “It is also difficult to backtrack+backspace because the error could be in the middle of the whole phrase at times”
29. “Taking longer to fix also it is very loud to speak as everyone can hear”

5.9.9.3 Like about keyboard

30. “Private/discreet”
31. “Can enter words with proper nouns. Irregular spelling and punctuation”
32. “The error correction/suggestions is pretty good”
33. “Accuracy is much higher compared to speech”
34. “Unusual nouns- for example uncommon names are easy to input because user has full control over keys + corrections”
35. “I tend to make lots of little errors in my keyboard typing and I like how it autocorrects for me”
36. “The predictive text was excellent”
37. “gives me time to think what next I would like to write”
38. “allows me to type any symbols and emoticons to express true feelings /thoughts”
39. “I was able to quickly backspace an error +also use autocorrect for some words”

5.9.9.4 Dislike about keyboard

40. “If your hands are occupied makes it hard to use keyboard”
41. “takes longer to type compared to speech”
42. “typing long sentences over a period of time fingers hurt”
43. “google keyboard wouldn't suggests words if you got at least one letter wrong”

5.9.9.5 Reasons for choosing speech

44. “I will prefer voice unless I'm in a library or a crowded place”
45. “allows to interact without having to focus solely on the phone i.e. crossing the road, you don't have to look at the screen”
46. “Most likely I will use it whilst driving using apple car play technology”

5.9.9.6 Reasons for choosing keyboard

47. “I will not use speech in real life. As I type fast, I feel my privacy is violated when using speech and if I can talk I'll call or send voice message”
48. “social (using speech in a social setting can be weird)”
49. “privacy (not everyone has to know my messages)”
50. “when I'm talking to someone I want to be clear with what I'm saying I cant take the risk of errors using speech to text”
51. “I don't know how punctuation works with speech”
52. “self-conscious about shouting out half of a conversation in public”

5.9.9.7 General comments

53. “Would be inclined to use speech more as an input method now”
54. “This experiment was wonderful and has made me curious as to how accurate the speech method is”
55. “surely, I learned that speech method is a very useful and I think I will be more likely to use it now than before”
56. “I'm still more likely to use the keyboard because I tend to text people if I'm in a place where I can’t call them to speech wouldn't quite help. ; I would definitely use speech, however, when I'm too lazy to type”

5.9.9.8 Analysis of open comments

We divided the comments into the follow categories

	Typing		Speech	
	Positive	**Negative**	**Positive**	**Negative**
Speed		41 (A)	1, 8, 10, 12, 13 (B)	
General Accuracy	33, 35, 36 (C)	43 (D)	2, 4, 5, 7, 8, 11 (E)	18, 20, 25, 26, 50 (F)
Convenience	37 (G)	40, 42 (H)	3, 6, 9, 10, 12, 13, 14, 44, 45, 46 (I)	
Editing/Correction	32, 34, 35, 39 (J)			21, 22, 24, 28, 29 (K)
OOV, Perplexity, Punctuation	31, 34, 38 (L)			16, 17, 19, 27, 51 (M)
Privacy & Social	30 (N)			15, 23, 29, 44, 47, 48, 49, 52 (O)

It can clearly be seen that users prefer (and would choose) typing over speech, or vice versa, for a largely different sets of reasons. What participants found positive about speech was mostly the speed, the general accuracy (when speaking commonly used sentences), and the convenience – a.k.a. not having full visual feedback, ability to focus on something else whilst doing it e.g. driving, and the ease of use – as seen in categories (B,E,I).

While the users complained about speech sometimes being non-accurate with particular accents (F) – most of the complaints were directed towards post-editing and correction (K), entering proper nouns, OOV words and long complex sentences (M), and privacy and social concerns (O). With typing, all the positive aspects were about it being discreet/private (N), the ability to enter anything you want, e.g. OOV's, long complex sentences (L), and the ability to edit and correct to your satisfaction (J). This gives us a good understanding of when users would opt to use keyboard or speech as a medium of input.

5.10 Summary

5.10.1 Entry Rates

In this table, we report the entry rates from the various analyses performed.

Mean Entry Rates	Normal	Excluding OOV Words	Only Sentences with OOV Words
Typing	40.23 (SD=12.27)	40.60 (SD=12.27)	31.44 (SD=8.51)
Speech	58.88 (SD=21.51)	59.22 (SD=21.38)	50.75 (SD=22.83)
Is difference significant?	Yes	Yes	Yes
Improvement over the sessions?	Yes	Yes	Yes

So this tells us that, in a lab setting, under a normal phrase set mixed between OOV and non-OOV words, speech clearly outperforms typing in terms of entry rate. Further, participant's entry rate will improve with time. Also, we can see that with or without the presence of OOV's, speech would still outperform typing in terms of entry rate – in a lab setting.

Therefore we have enough evidence to reject the following null hypotheses:

H0,a There is no difference in text entry rate between the keyboard and speech in a lab setting

And replace them with the following alternate hypotheses:

H1,a Speech has a faster entry rate than typing in a lab setting

However, there is not enough evidence to reject the following null hypothesis:

H0,c The presence of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of entry rate in a lab setting

5.10.2 Error Rates

In this table, we will consider only Character Error Rates (CER) as this was the basis for the statistical analysis. Word Error Rates were observed to follow a similar pattern with higher values.

Mean Error Rates	Normal	Excluding OOV Words	Only Sentences with OOV Words
Typing	0.22% (SD=0.99)	0.20% (SD=0.94)	0.70% (SD=1.69)
Speech	3.34% (SD=6.84)	3.07 % (SD=6.56)	10.09% (SD=9.54)
Is difference significant?	Yes	Yes	Yes
Improvement over the sessions?	Yes	No	Yes

So this tells us that, in a lab setting, under a normal phrase set mixed between OOV and non-OOV words, typing outperforms speech in terms of error rate. Further, participant's error rate did not change with time for sentences without OOV words. When using sentences with OOV words, the participant's error rate varied with block.

Therefore we have enough evidence to reject the following null hypothesis:

H0,b There is no difference in character error rate after correction between keyboard and speech in a lab setting

And replace them with the following alternate hypothesis:

H1,b Speech produces more errors than typing in a lab setting

However, we do not have enough evidence to reject the following null hypothesis, as with and without OOV words, there is still a significant difference in error rates between the two input methods.

H0,d The presence of of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of error rate in a lab setting

5.10.3 Sentence Perplexity

In this table, we summarise the results from the hand posture analysis.

	Significant Effect of Input Method	Significant Effect of Quartile	Interaction between IM x Q
Entry Rate	Yes	Yes	Yes
Error Rate	Yes	Yes	Yes

This, combined with the above results, clearly tells us that speech is affected more in terms of both entry and error rate when the sentence perplexity varies. Higher perplexity results in a much higher error rates and slower entry speeds in speech.

Therefore, we can reject the following null hypotheses:

H0,g The perplexity of the phrase does not affect keyboard and speech differently in terms of entry rate, in a lab setting

H0,h The perplexity of the phrase does not affect keyboard and speech different in terms of error rate, in a lab setting

And replace it with these alternate hypotheses:

H1,g Sentence perplexity affects speech more than keyboard in terms of entry rate in a lab setting

H1,h Sentence perplexity affects speech more than keyboard in terms of error rate in a lab setting

5.10.4 Subjective Ratings & Open Comments

From the results and analysis above, there is not enough evidence to reject the following null hypotheses:

H0,e The user experience of the participants did not differ between keyboard and speech
in a lab setting

H0,f In the lab, users did not prefer to use one method over the other when given a choice between both

As there was no significant difference in the self-ratings that users provided for perceived input speed, perceived accuracy, perceived ease of use and preference.

5.11 Conclusions

All in all, we could draw the following conclusions from this study:

H1,a Speech has a faster entry rate than typing in a lab setting

H1,b Speech produces more errors than typing in a lab setting

H0,c The presence of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of entry rate in a lab setting
-- *not enough evidence to say otherwise*

H0,d The presence of of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of error rate in a lab setting
-- *not enough evidence to say otherwise*

H0,e The user experience of the participants did not differ between keyboard and speech
in a lab setting

H0,f In the lab, users did not prefer to use one method over the other when given a choice between both

H1,g Sentence perplexity affects speech more than keyboard in terms of entry rate in a lab setting

H1,h Sentence perplexity affects speech more than keyboard in terms of error rate in a lab setting

6

Study D – Comparison of Typing and Speech in a the Wild

6.1 Motivation

Even though speech has joined the mainstream text entry methods on a plethora of devices, the HCI research literature offers little empirical evaluation of the current state of affairs in general, and the performance and experience difference between keyboard and speech in particular. Empirical research has been limited in scope, size, and technology form factor. Most reported text entry research has also been based on research prototypes. Continued progress and innovation in the text entry field cannot have a solid empirical footing if we do not know how well current technologies work for users. Further, despite the prevalence of both methods there is a lack of in-depth studies about their text entry performance, in particular outside a lab environment. Last but not least, most literature focusses on niche aspects of speech such as correcting errors, predicting words better, and was done before speech recognition had its massive improvements in the last few years, due to increased computational power, and breakthroughs in machine learning.

In this chapter, we empirically compare two state-of-the-art text input methods outside the lab – in a real world setting, which we call "in the wild". This is especially interesting as we evaluate the system under a variety of circumstances, which will be described in the following sections.

6.2 Hypotheses

We present the following null hypotheses which are to be accepted or rejectd as a result of this study. As shown, this study is broad and sheds light on many different aspects of text input between the two keyboards.

H0,a There is no difference in text entry rate between the keyboard and speech in the wild

H0,b There is no difference in character error rate after correction between keyboard and speech in the wild

H0,c The presence of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of entry rate, in the wild

H0,d The presence of of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of error rate , in the wild

H0,e The user experience of the participants did not differ between keyboard and speech in the wild

H0,f Users did not prefer to use one method over the other when given a choice between both, in the wild

H0,g The perplexity of the phrase does not affect keyboard and speech differently in terms of entry rate, in the wild

H0,h The perplexity of the phrase does not affect keyboard and speech different in terms of error rate, in the wild

6.3 Variables & Confounds

In this study, we identify three types of variables as independent variables, dependent variables, and confounds.

6.3.1 Independent Variables

These are the variables we explicitly control in this study.

- V1 – Input mechanism (2 levels: Keyboard and Speech)
- V2 – Participant (12 levels: P1-P12)
- V3 – Block (5 levels: B1-B5) – the data divided into 5 blocks based on chronological order
- V4 – Phrase Perplexity (4 levels: PPL1-PP4)

More details on phrase perplexity will be discussed in the Materials section under Phrase Set.

6.3.2 Dependent Variables

These are the variables that we measure as an outcome of this study. The measurements lead to "derived dependent variables" which lead to the analysis of the study results. This means we do not measure these directly but we derive them via calculations from the dependent variables we measure. The following sub sections below describe the variables we measure vs the variables we derive.

6.3.2.1 Measured Dependent Variables

These are the dependent variables we explicitly measure.

6.3.2.1.1 Timestamp at start of entry (T1)

In the case of keyboard, this is the timestamp when the user first begins to type. The time T1 indicates exactly when the first keystroke – or in this case, when the users finger touches the area surrounding the keyboard. On the Android platform, this is normally captured with an *onKeyDown* event.

In the case of speech, this is timestamp when the user presses the SPEAK button, which activates the microphone and speech widget, using the Android Speech Recognition API (Google, 2018d)

6.3.2.1.2 Timestamp when finished entering text (T2)

Theoretically, this is the timestamp when the user enters the last character in the sentence or phrase they intend to type, in the case of keyboard, or when the user stops speaking, in the case of speech.

However, this is impossible to capture programmatically as there is no way a program can know when the user has finished typing – i.e. the key the user just pressed could the last one, or there could be more to come. In speech, this could be captured as when the mic is deactivated (upon a significant spell of silence), however, we do not know if the user wishes to enter more text using speech by reactivating the microphone again.

Therefore when running studies, we use a practical delimiter to capture when the user has completed typing – such as pressing a button which says NEXT, or FINISHED, or performing some other delimiting action, which tells the program that the user has indeed finished typing or speaking. In a texting application this would be denoted by pressing SEND. In this study, we capture the "end of phrase" when the user indicates they want to move to the next sentence by pressing NEXT.

It is obvious that there is a slight delay between entering the last character in the response phrase and pressing next, however, this does not skew the results in the study as:

n. When typing, this happens almost instantaneously
o. When entering text via speech, the user presses NEXT almost instantaneously after getting what they need
p. We explicitly tell the users to use a minimal delay between finishing entering text and pressing next
q. This delay is uniform across the entire study (and does not differ between subjects)
r. If the user does require to proofread what they entered, this should be indeed factored in to the time it takes to enter text using the given input mechanism, as this is a critical factor

6.3.2.2 Derived Dependent Variables

These are the dependent variables we calculate from the measured dependent variables. Descriptions of these can be found in Chapter 1 – Introduction, under Conventions.

- Number of characters in the response phrase (N)
- Typing duration (T)
- The Error (E)
- Entry Rate (WPM)
- Error Rate

6.3.3 Confounding Variables

These are variables that we did not try to control, but still would be consider as variables due to their confounding nature, as they can definitely affect the typing experience and performance in the study. We identify one confounding variable in this study - description of which can be found in Chapter 1 – Introduction

- OOV words

6.4 Apparatus

This section explains the hardware the software apparatus used for the study.

6.4.1 Hardware Apparatus

For this study, we used the participants own devices. The rationale for this is that when performing studies “in the wild” the participants must have their phone with them the whole time. If the authors were to provide participants with a mobile device, this would be unrealistic for two reasons:

(c) This would not be their primary phone – therefore collecting data via this device would not yield accurate data pertaining to their actual behaviour

(d) The participants will not be familiar with the device, therefore the data will be unrealistic

(e) Their speech recognition is personal to their Google account, therefore it would yield the most realistic results when they are logged into their own Google account from their own device

To ensure that the software apparatus runs properly, and there's not too much difference between the device form factors, we filtered the participants based on the specs of their primary mobile device. The criteria were:

(c) They must have Android 7.0 or later (most participants had 8.0)

(d) Should contain the Google Play Services and Voice Commands Capability (for speech to function properly)

Upon screening and selection, the resultant devices used for the experiment were as follows:

Phone Make & Model	Form Factor	Android Ver.	#
Samsung Galaxy S7	5.1"	8	1
Samsung Galaxy A3	4.5"	7	1
Samsung Galaxy S9	6.2"	8.1	2
Lenovo Moto G4	5.5"	7	1
Lenovo Moto G5	5.0"	7	1
Sony Xperia XA1	5.0"	8	1
Samsung Galaxy S6	5.1"	7, 8	2
OnePlus A3	5.5"	8	1
OnePlus A6	6.2"	8	1
Huawei P20 Lite	5.8"	8	1

6.4.2 Software Apparatus

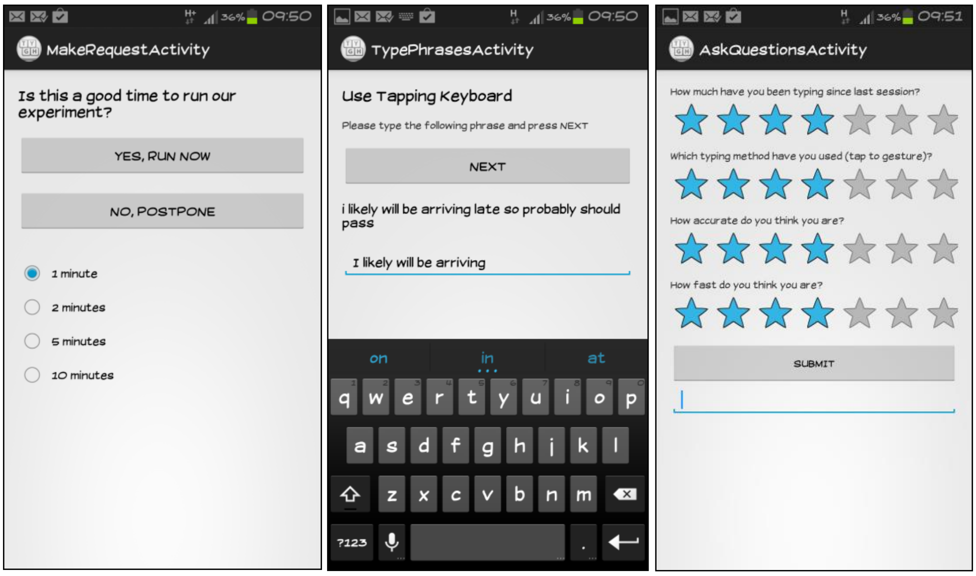
There were two major components in the software apparatus. The First was the Google Keyboard, which had its own implementation of state-of-the-art Keyboard built in. The second was the experimental software that was required to run the study, into which we built an implementation of the AndroidSpeechRecognizer (Google, 2018d) The participants were required to download and install the Google Keyboard and set it as their main method of input, and the experimental software, described as follows.

6.4.2.1 Experimental Software

The app used for the study outlined in this chapter is designed for an ESM study carried out in the wild. It was important that the app handled everything in a fully automated manner with minimal or no experimenter intervention. The participants could find themselves in any situation during the study and the app had to be ready to deal with all these foreseen and unforeseen circumstances.

The app has a basic start up screen for entering information such as participant ID. Once started, the app will be working in the background for the duration of the experiment (e.g. 1 month). At pseudo-random intervals of the day (roughly spaced apart by 1 hour), the app will come to the foreground and request the user to perform a task. The task is a transcription task where the user is shown a stimuli sentence and requested to copy it using one of the methods (STK or SGK). However, the user mind find themselves in a situation where this is not possible, in which case they can choose to “snooze” this request for 1, 2, 5 or 10 minutes, after which the app will remind them again.

This behaviour was achieved by using the Android’s AlarmManager class. If the user decides to accept the request, then the user will be shown a basic screen with a textbox and next button. The user simply has to copy the sentence given, using the condition (see image – shows “use Tapping Keyboard”) and press Next. Once they do, another basic screen will show them 4 questions with 7 point Likert scales as responses (see materials), these are to capture the users perceived performance and comparative performance (in comparison to the previous session).


MakeRequestActivity

Is this a good time to run our experiment?

YES, RUN NOW

NO, POSTPONE

● 1 minute
○ 2 minutes
○ 5 minutes
○ 10 minutes

TypePhrasesActivity

Use Tapping Keyboard

Please type the following phrase and press NEXT

NEXT

i likely will be arriving late so probably should pass

I likely will be arriving

AskQuestionsActivity

How much have you been typing since last session?

Which typing method have you used (tap to gesture)?

How accurate do you think you are?

How fast do you think you are?

SUBMIT

Figure 63 - Software Apparatus used in Experiment D

The stimuli phrase shown here is from a randomized copy of a given data set (we use the Enron Mobile Email Dataset for this study – see Materials), shuffled using the Fisher Yates shuffling algorithm (Fisher & Yates, 1948). The two timestamps where the user enters the first character and presses the NEXT button are also captured. The stimuli and response phrases, as well as the timestamps are written to files in the mobile's internal storage. Once this process is complete, the app goes to the background again and waits until it is time to come up.

Once the mobile connects to a WiFi network, the app identifies this and uploads the data it saved to a specified URL via a HTTP POST request. A server side application was implemented to capture this request and read the saved data from it in XML format. The app handles any anomalies that could occur when this network transfer is taking place.

Since this app is supposed to work "in the wild", there are so many unforeseen situations that the app must be made ready to, unlike in a lab experiment. Examples for such situations are:

- The phone can run out of battery and die. This can be between requests, or while the participant is actually servicing a request. The experiment should commence when the phone turns on again, from where it stopped, so the participant doesn't need to redo what they had done before.

- The phone can be rotated – and due to the Android implementation, the entire android activity is killed and recreated, thus clearing all temporary and global variables stored in the app. Therefore every single action pertaining to saving state or data should always be stored in persistent storage and retrieved.

- The user may forget or ignore to provide a response at all, in which case the app must keep reminding the user. This is implemented via a tolerance duration variable (i.e. 15mins), after which the app will buzz again.

- The user might want to set the app inactive during the times he/she sleeps. This is implemented via a "quiet time" setting, where the app can be configured not to buzz a user at certain times of the day.

- The user might accidently press the home button and send the app to the background while typing and the app should recognize this and come to the foreground again – either immediately or after the tolerance duration mentioned above. The back button has been disabled for similar reasons, but the home button cannot be disabled without rooting the device.

It should be noted here that the app does not collect any data outside these experimental sessions, thus not raising any ethical or privacy concerns for the participants. This app is indeed a very useful tool for researchers who wish to conduct such experiments without having to “reinvent the wheel”.

This application was developed using Java/Eclipse, and could be deployed on any android mobile device running Android version 4.0 or later - the requirement for version 4.0 was to align with the requirements of Google keyboard. It can be deployed as a single APK file on a mobile device with ease.

6.4.2.2 Software Design

This application contains 3 parts which can be treated as pluggable modules which interact with each other.

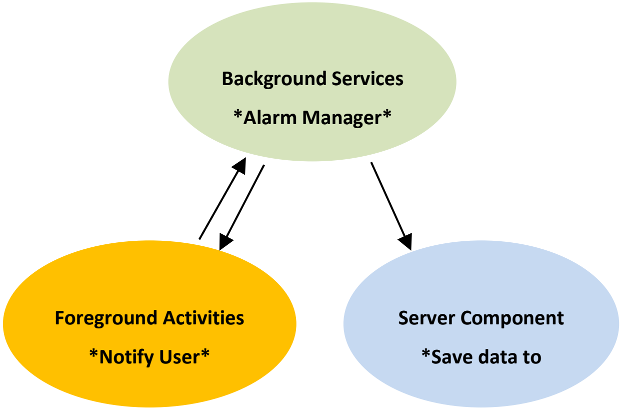


Figure 64 - High Level Component Diagram

The entire operation of the app can be simplified into the diagram below. In addition to this we built a simple server application which reads data from an HTTP request and writes it to a database. We implemented ours using PHP and MySQL, but it only requires to be a simple application that listens to Http Requests and reads XML.

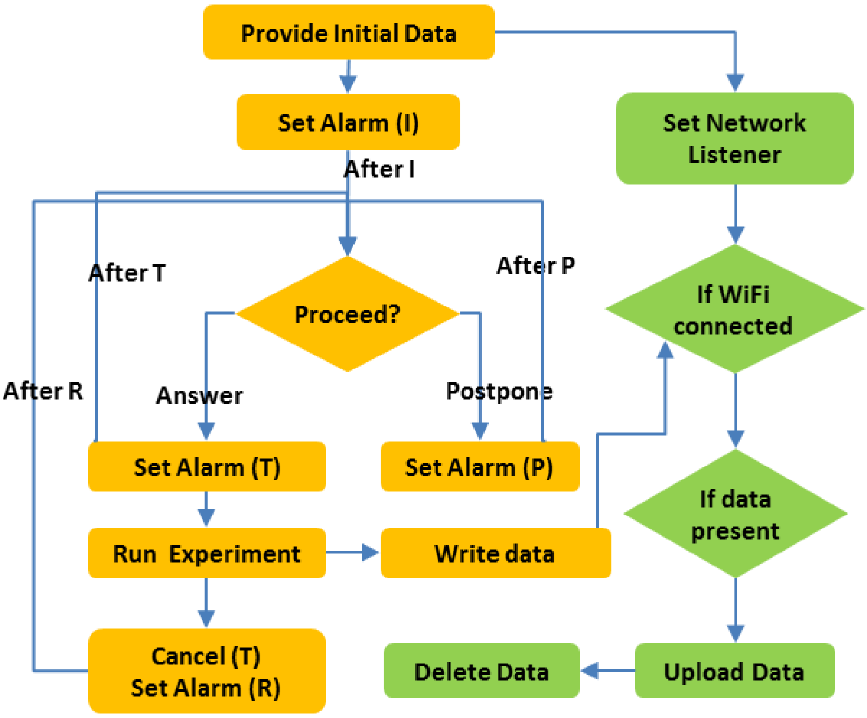


Figure 65 - Operational Flow Diagram

All the configuration parameters are defined as constants therefore anyone could customize the application to their requirement such as the data set used (i.e Enron), number of runs (i.e. 300), number of phrases to type per run (3), delay values, I, T, P, R, etc.

6.5 Materials

This section describes the surveys used, the phrase set used for the above study, compensation and phrase set.

6.5.1 Surveys

In addition to capturing the user's performance when using either type of keyboard, we also surveyed their responses on previous typing experience, mobile phone experience, smartphone experience, and perceived performance and user experience, which shall be explained in the upcoming sub sections.

6.5.1.1 Preliminary Survey

This was given at the beginning of the entire study, before the user had the opportunity to enter any text whatsoever. The purpose of the survey was to gauge the prior experience of the user.

Q1. In your life, which text entry method did you use more during your day-to-day activity?

Only Tapping 1 2 3 4 5 6 7 Only Gesturing

Rationale (Q1):

This, in conjunction with the final survey (presented at the end of the study), this will reveal if users had prior experience with gesture keyboard, and if not, would have started using gesture keyboard in their day to day life as a result of their study.

Q2. What kind of mobile devices do you use? Tick all that apply.

Smartphone – Android
Smartphone – Apple
Smartphone – Microsoft
BlackBerry
Feature phone (no large touchscreen)
Tablet – Android
Tablet – Apple
Tablet – Microsoft
Phablet – Android (A very large smartphone)

Q3. Please write the brands and models of the mobile device you have used the most in last few years.

Brand	Model	Duration of Use

Q4. Please rate your ability to type using your mobile device.

Very slow typist	1	2	3	4	5	6	7	Very fast typist
Inaccurate typist	1	2	3	4	5	6	7	Very accurate typist

Rationale (Q2-Q4):

This is to gauge the user's previous smartphone experience, which is directly attributable to one's performance on a STK and SG. This will be revisited in the participants section.

6.5.1.2 Surveys during the studies

At each experience sample, we gauged the participant's "in-situ" user experience four questions, response for each of these being a Likert Scale from 1-7.

Q5. How much have you been typing since last session?

Very Little 1 2 3 4 5 6 7 All the time

Q6. Which typing method have you used outside this study?

Only Tapping 1 2 3 4 5 6 7 Only Gesturing

Rationale (Q5-Q6):

We wanted to find out how much the users have typed outside this study (between each experience sample), just to see if that has a correlation with their performance during the sample

Q7. How accurate do you think you are?

Not at all 1 2 3 4 5 6 7 Very Accurate

Q8. How fast do you think you are?

Not at all 1 2 3 4 5 6 7 Very Fast

Rationale (Q7-Q8):

These were regards to the experience sample they just completed; we wished to find out how they perceived themselves in terms of being fast or accurate.

6.5.1.3 Final Survey – End of the study

This was the final questionnaire at the end of the full study. We required the participants to provide both quantitative and qualitative/open ended answers on what they liked and disliked about each input method. These questions were based on the entire study experience a.k.a. the full 1 month duration.

Q9. During this study, I was mostly

Stationary 1 2 3 4 5 6 7 On the move

Q10. During the study, I was mostly

Energetic 1 2 3 4 5 6 7 Exhausted

Q11. I most actively interact with my mobile device during

Morning 1 2 3 4 5 6 7 Night

Q12. My general method of text input before the experiment was

Tapping 1 2 3 4 5 6 7 Gesture

Q13. My general method of text input after the experiment is

Tapping 1 2 3 4 5 6 7 Gesture

Q14. When the app made a request when I was walking, when entering text I

Stopped 1 2 3 4 5 6 7 Continued walking

Rationale (Q9, Q14):

As the question suggests, we wanted to find out if the users were mostly stationary or on the move during the study. This is simply to obtain an idea of what the circumstances we sampled their experience in.

Rationale (Q10-Q11):

We wanted to find out if fatigue, exhaustion, energy levels, time of day affected our participants ability to type during the study.

Rationale (Q12-13):

This was to find out if the study had actually affected the users day-to-day typing. As some users had not been exposure to gesture keyboard before this study, we wanted to find out if they used it regularly.

Q15. What did you like about each input method?

Q16. What did you dislike about each input method?

Q17. Did your everyday text input get affected as a result of this experiment?
Did you learn a new method of text input (i.e. gesture input), became aware about a new tool (i.e. Google keyboard), apply it to your own day-to-day life, or did you become faster, more accurate etc.

Q18. What features in the application did you find desirable?

Q19. What features in the application did you find un-desirable?

Q20. What improvements would you suggest we do for this app if we plan to run the study again?

Q21. What do you think about this experiment?

Q22. What was the hand posture that you used for each typing method (pick the most used).
TAPPING – Thumb | Two Thumbs | Single Finger
GESTURE – Thumb | Two Thumbs | Single Finger

Rationale (Q15, Q16):

We wanted to capture which aspects of each input method they like and disliked, as this would give us insight into what features work better and when

Rationale (Q17):

This was to strengthen and justify the values in Q12 and Q13

Rationale (Q18-Q21):

This was simply to find out what could be improved with the study. We did not use this information in the results part of this thesis, yet the suggestions made here were used to improve the studies (see Chapter 6)

Rationale (Q22):

We wanted to gauge the participants hand posture during this study when using STK and SGK

6.5.2 Phrase Set

We used a subset of the Enron mobile email dataset (Keith Vertanen & Kristensson, 2011) with the following conditions:

- Each sentence should be less than 60 characters in length
- No numbers
- No special symbols
- Sentences with varying perplexity from 1.68941 - 13.518

This resulted in phrase set of 1016 phrases. We counted 1382 unique words in this test set. The rationale behind this was we didn't want users to switch between keyboards to enter numbers and special symbols – i.e. in Android, when using Google Keyboard; users have to change the view back and forth to enter numbers, symbols and letters.

Further, there is no guaranteed way in speech recognition to enter punctuation marks or numbers. I.e. if the user speaks "two" the speech engine might infer the word "two" or the number "2", similarly with punctuation i.e. if the user speaks "question mark" it may result in the word or the symbol. Therefore we decided these to be explored in a different study.

6.5.2.1 OOV Words

We did not, however, exclude sentences with Out of Vocabulary words (OOVs). We did this because we want to find out how each input mechanism performed differently when OOV words were part of the mix – especially speech. As speech has no way of inferring OOV words, it was an interesting observation as to how it affected the typing speed, the error rate, the user experience and most of all how users dealt with this particular issue when typing.

We compared all the words in the phrase set against a standard lexicon (64K common words used in the English language). The words that weren’t in the lexicon were each entered carefully on the Google keyboard, by tapping the center of each key on the STK, by gesturing from the center to center of each key on the SGK, and clearly pronouncing the word into the voice input of Google keyboard multiple times. We noted that the same 39 words were out of vocabulary (OOV) words for STK, SGK, and speech. These OOVs appeared in 40 sentences (3.94% of 1,016) in the phrase set, and were marked as sentences with OOV words. These OOV sentences were analysed in post-hoc analyses after the experiment.

6.5.2.2 Non OOV Samples

The following are a few sample phrases from the set which do not include OOV words

'Any time Thursday'
'Can we have them until we move'
'I compliment you'
'Jan has a lot of detail'
'Still waiting on decision'

6.5.2.3 OOV Samples

The following are a few sample phrases from the set which include OOV words

'Have we assigned employees to NetCo'
'He will walk Tanya Rohauer through the exact same steps tomorrow'
'If so Whitt is done'
'It should be Cynthia Barow instead'
'John Keffer is the one I know best'

6.5.2.4 Sentence Perplexity

We wanted to find out if sentence perplexity affects speech and keyboard at the same level, therefore we calculated the perplexity of each phrase in the chosen set and divided them into 4 quartiles as follows.

Quartile	Min PPL	Max PPL	Example Sentences
Q1	1.68941	2.33612	I plan to be in the office tomorrow (2.13943)
Q2	2.33715	2.68698	Are you in a good mood (2.48629)
Q3	2.68758	3.14593	This is the crew (2.90663)
Q4	3.14939	13.518	He will walk Tanya Rohauer through the exact same steps tomorrow (3.62079)

6.5.2.4.1 Calculating Sentence Perplexity

We calculated the perplexity of these 1016 Enron mobile sentences under a 12-gram language model trained on billions of words of mobile-like data. We limited to sentences with between 3 and 12 words. The average Turk worker memorization CER had to be between 0 and 10%. We excluded sentences with punctuation other than apostrophe and end of sentence punctuation, and sentences with the digits 0-9. We also removed sentences with words of 2+ letters in capitals (usually abbreviations). The ARPA format gzipped language model is 1012MB in size (it is only lightly pruned). The vocabulary size is 34: A-Z plus .,'! and pseudo-word for space.

6.5.2.5 Ordering of Phrases

When the phrases were displayed to the participants during the study, they were chosen at pseudo-random – every 4 sentences shown to the participant were chosen at random from each of the Quartiles above, ensuring that each participant on each session entered an equal number of sentences from each Quartile/ The Fisher Yates shuffling Algorithm (Fisher & Yates, 1948)was used inside each quartile to randomize each subset, and then was simply merged together using a “merge in turn” algorithm. The two algorithms can be outlined as follows.

```
// ar is the array that needs shuffling

public static void fisherYatesShuffle(int[] ar) {
    // generate a randomizer
    Random rnd = ThreadLocalRandom.current();

    for (int i = ar.length - 1; i > 0; i--) {
      // generate random index between 0 and (i+1)
      int index = rnd.nextInt(i + 1);

      // perform a simple swap between the current and
      // random positions
      int a = ar[index];
      ar[index] = ar[i];
      ar[i] = a;
    }
}
```

```
// ar1, ar2, ar3, ar4 are 4 equally sized arrays
// each representing one randomized quartile of the phrase set

public static int[] mergeInTurn(int[] ar1, ar2, ar3, ar4) {
    int[] result = new int[ar1.length * 4];
    int index = 0;

    for (int i=0; i<ar1.length; i++) {
        result[index++] = ar1[i];
        result[index++] = ar2[i];
        result[index++] = ar3[i];
        result[index++] = ar4[i];
    }

    return result;
}
```

6.6 Compensation

Participants were compensated £30 for their time in amazon vouchers. Further we offered an incentive of an extra £15 for the fastest typing participant in each keyboard type, under a certain error rate threshold.

6.7 Participants

We recruited 12 volunteers from the university campus. As per previous studies, the participant number was justified by previous studies performed in literature (P. O. Kristensson & Denby, 2009). 5 participants were from University of Cambridge, and 7 were from University of St Andrews. Again these too were a rather broad sample as they came from various schools and departments.

6.7.1 Participant Demographics

Due to the ethics agreement we cannot publish any identifiable information about the participants - therefore the aggregate results of each demographic will be described below and not attributed to individual participants.

6.7.1.1 Gender

We had an equal number of males and females, 6 each.

6.7.1.2 Age

Their ages ranged from 21 to 37, with a mean of 29.3 and Standard Deviation of 4.6.

6.7.1.3 English Proficiency

Two of them had English as their first language whilst the others practiced English as their second language. As per the previous experiment the participants used English regularly for studies and conversation. Given they were either doing their masters or PhD, or employed in University of St Andrews or University of Cambridge, they had to be proficient in English – as per the English language requirements of university admissions. This ensured that our participants were able to understand, read and copy the sentences in the above phrase set without difficulty. Most of the participants used the English language in their day to day life for exchange of messages over mobile devices except for two of them who were using a keyboard with English keys but output transliterated Chinese characters. They were screened for their competency in English and proved to be satisfactory.

6.7.1.4 Geographic Distribution

The University of St Andrews and University of Cambridge attract a very diverse student and staff demographic, and the 12 participants were distributed across the globe as follows.

The participants also self-assessed their accents, as we were doing a study on speech recognition – in case it might come in handy for the subsequent analyses.

Country	#	Accent (self-assessed)
England	2	English
Italy	1	Italian
Hungary	1	Eastern European
Romania	1	
Pakistan	1	Indo-Pakistani
Sri Lanka	1	
Bangladesh	1	
India	1	
Brazil	1	Hispanic
Argentina	1	
Spain	1	

6.7.1.5 Field of Study Distribution

The participant's field of study varied across disciplines as follows

Computer Science	Chemistry	Management
7	4	1

Further, the participants' level of study or career can be summarised as follows.

Masters	PhD	Academic Staff	Professional Staff
1	5	3	3

6.7.1.6 Smartphone Experience

All the participants owned Android devices running version 7.0 or later, with Google Keyboard.

6.7.1.7 Exposure to Typing and Speech

Again by interviewing the participants, we gauged their previous experience with typing and speech. They self-rated themselves on how much they used either typing or speech, and in the case of typing, whether they used STK or SGK in their day-to-day lives.

Further, the participants also self-rated their favourite or most-used hand posture when typing on mobile devices. In the graph below, Typing or Speech is a rating on 1-7 Likert Scale, where 1 is mostly typing and 7 is mostly speech. In the same way, STK or SGK is also a 107 Likert Scale, where 1 is 100% STK usage, and 7 is 100% SGK usage.

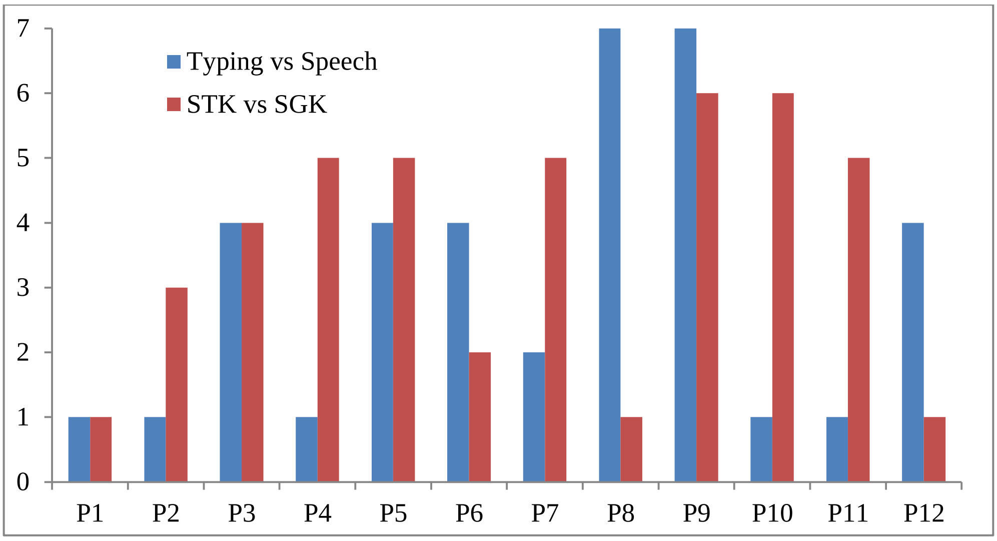


Figure 66 - Users previous experience with typing, gesturing, and speech

The participant's usage on different hand postures were as follows:

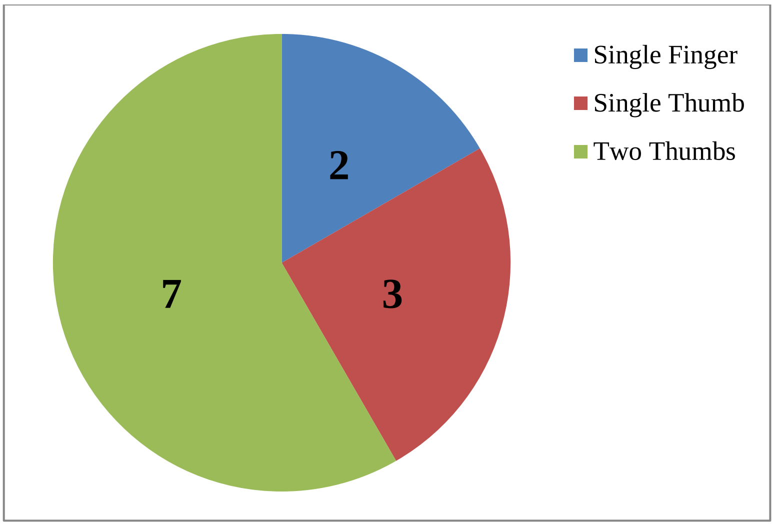


Figure 67 - Users previous experience on hand postures

6.7.2 Study Design

A within-subjects design was intended for this study - meaning each participant had to experience both conditions (both typing and speech). To minimise starting bias, we had 6 participants use STK first, and the remaining 6 participants to use SGK first. This ensured that we had a balanced group of participants with fully balanced conditions between the two groups.

6.7.3 Recruitment

This section explains how participants were recruited and screened. The sample advert was as follows

Participants needed for a study on Text-Input.
Compensated £30 + Chance to win extra £15 in Amazon Vouchers.

We invite you to participate in a PhD research project examining how typing speed and error rate changes when entering text on an Android mobile device with two different input methods (Typing Vs. Speech). We will be using the Google keyboard which supports normal tapping of keys on a virtual (soft) keyboard and also continuous input for gestures.

The only requirements for participation are that you are above 18, used to entering English text on a mobile device, do not suffer from any learning or communication disabilities, and own an android mobile device.

Each participant will be required to install Google keyboard (if not already present) and a custom application on their mobile device. The application will run in the background for 2 weeks. The application will come to the foreground at random times of the day (~30 occurrences a day) and ask the user to enter 4 phrases. The app will record the typed phrase, input speed and error rate, as well as motion sensor data from your mobile to identify whether the participant was still or moving.

When the user connects to Wi-Fi network, the aforesaid recorded data will be sent back to a server where it will be collected. The application will not record any other data while running in the background.

You will be reimbursed £30 for your time, provided that you follow the experiment instructions. Two participants who fare the highest speed and accuracy in each method of input will be awarded an extra £15 each. All payments will be made in the form of Amazon Vouchers.

If you are interested in participating please contact Shyam Reyal on
smr20@st-andrews.ac.uk or 07447924147.

Participants were screened for:

(j) English proficiency – either they had to be native English speakers or use English as a second language in their day to day life / studies

(k) Experience with QWERTY – since we use this keyboard layout, participants had to be experienced in typing using a QWERTY keyboard either using the computer or their mobile device

(l) Type of mobile device – they had to own an Android mobile device running version 7.0 or later, and had to have the Google Play Store available

We used various channels to advertise this study and recruit participants, briefly outlined below.

6.8 Procedure and Execution

The execution of the study was done in two steps. The first was a pilot where the authors of the paper used the apparatus to find any errors in the implementation that could affect the study, followed by the actual study involving the 12 participants.

6.8.1 Pilot Study

The study apparatus was piloted intensely before the actual study commenced, and a number of problems were identified. The software apparatus was based on the same apparatus which successfully ran in 2013, however, due to the changing nature of Android API's most of the code written, tested and verified to work had to be rewritten.

6.8.1.1 The Non-Guaranteed AlarmManager

The Android **`AlamManager.setRepeating(interval)`** was no longer guaranteed to trigger an event at a set time. Due to differences in platforms, and updates to the Android operating system, the times passed into "interval" above could be overridden by the operating system to optimize battery etc. Therefore during piloting we found that the app stays in the background for hours and not trigger every 15-20 minutes as intended.

The solution for this was to replace **`setRepeating`** with **`setExact`**. This proved to be better solution when testing the app, but still was not guaranteed, as Samsung-Smart-Manager^{TM} would override this setting to optimise battery and memory usage.

6.8.1.2 The Silent Vibration

In the 2013 version of the app, the users were aggressively reminded with a tone + vibration, (or just vibration, if the phone was on silent). It used to be impossible to turn this vibration off if the app was requesting it. However, with the new versions of Android, the vibration is also not guaranteed. Therefore, if the user put the phone on silent, they might have no way of knowing the app is requesting them to run the study.

6.8.1.3 Apps that require Overlay

This was particularly seen on Huawei mobile devices – where the manufacturer supplied OS completely suppresses any apps that require overlay – a.k.a. coming to the foreground from the background. The app had to be granted special permissions from the app-settings for this to be enabled explicitly.

6.8.1.4 Runtime Permissions

The permissions requested via the Android Manifest file (in 2013), had to be replaced with Runtime Permissions, where the user has to explicitly grant permissions to the app upon installation. This was found out when one pilot participant was doing the study yet nothing was recorded or sent back, which shed light on the existence of this problem.

6.8.1.5 Google Voice Commands

Speech was introduced to the ESM app, and was working fine on many of the devices, however one participant had to explicitly install “Google Voice” on their mobile device before they were allowed to enter text using speech using the app.

6.8.1.6 Generation X Participants

We found that these participants had a different mobile phone usage pattern than the participants who ran the STK vs SGK ESM study in 2013. Whereas the previous participants (see chapter 4) used their mobile continuously throughout the day, these participants did not. They mostly had their mobile devices on silent for long stretches of time, and checked their phones for updates during ether lunchtime, or after work, or upon waking up in the morning. We identified this problem might be because of the participant age difference, and their work – i.e. the previous participants were mostly undergraduate and masters students, whereas the new participants were PhD students, academic staff, or professional employees.

To overcome this problem, we developed the app to be more aggressive – i.e. ping the user every 20-30 minutes as opposed to one hour during peak times – as defined by the users – and every 10 -15 minutes during off peak times – which was mostly weekends, and weekday evenings.

6.8.2 Actual Study

This procedure was fairly straightforward. The participants were given the custom built application to install on their mobile phones. The app ran in the background and from time to time asked the users to enter three phrases from the modified Enron mobile email data set. The users had the option of either responding immediately or postponing the request by either 1, 2, 5 or 10 minutes. If chosen to postpone, the app would remind them again in the designated time.

The experiment used a transcription task where participants were shown a phrase from the dataset and asked to copy it. Each participant was given a mobile application to be used over duration of 2 weeks, on which they would enter text during random times of the day. Each participant was compensated with £30 for their commitment in the form of amazon vouchers. To encourage the participants to enter text as quickly and accurately as possible, a reward of extra £15 was announced to the participants who fare the fastest and most accurate.

Whilst being encouraged to use the Google keyboard’s suggested words for correction, we discouraged participants to go back and correct errors unless absolutely necessary. I.e. when transcribing the phrase, if an error or typo was made, we asked the participants to decide if they would send this message off if a human user was listening on the other end. If they decided the original word was still able to be deciphered by the recipient, then they were not required to go back and correct it. Our experiment app recorded the stimulus phrases and the response text using millisecond timestamps when the user entered the first character and when the user pressed NEXT, in the case of keyboard, and when they started and stopped speaking in the case of speech. Participants rated their previous experience with software keyboards (STK and SGK) on mobile devices, and self-rated themselves on how fast and accurate they thought they were.

It should be noted here that the app did not collect any data outside these experimental sessions. We had timed the application so that the user would get one request every 20 minutes which are nearly equally spaced out during the time of day which the user is awake and is capable of interacting with his/her mobile device. Once the user has entered four sentences, the app goes to the background following a very brief questionnaire where the user has to reflect on his/her overall daily typing experience.

6.9 Results & Analysis

We ran the experiment for an approximate duration of two weeks. However, due to the problems mentioned beforehand (see Pilot Study), the participants did not complete enough data points on each conditions (Typing and Speech) for the study to be analysed as a within subjects design. However, given that the participants finished at least one condition (Typing or Speech) and we had an equal number of participants begin with either method to balance the conditions, we were able to analyse this study as a between subjects study instead.

At the end of two weeks, each participant entered the following number of sentences in their respective conditions. As shown here, certain participants (P4 – P9) had begun their second condition, yet do not have enough data points. Therefore, we will only consider the conditions (methods) which they have completed and assign the participants to a single condition – all subsequent analyses will assume a between subjects design where P1 – P6 are allotted to Typing and P7 – P12 are allotted to speech. They grey cells above include the data points which were ignored for the analysis of this study.

Data Points	Typing	Speech	Allotment
P1	164		Typing
P2	427		Typing
P3	139		Typing
P4	316	47	Typing
P5	423	8	Typing
P6	388	48	Typing
P7	207	421	Speech
P8	136	320	Speech
P9	36	424	Speech
P10		100	Speech
P11		116	Speech
P12		399	Speech

6.9.1 Data Preparation

Similar to the previous experiments, we prepared the data by removing outliers that were more than 3 standard deviations away from the mean w.r.t to character error rate. Upon removing the outliers, for typing, each participant had an average of 338 data points (SD=128), and for speech, each participant had an average of 297 data points (SD=151). This resulted in a total of 1857 data points for typing and 1780 data points for speech. The scatter plots for the data are shown below.

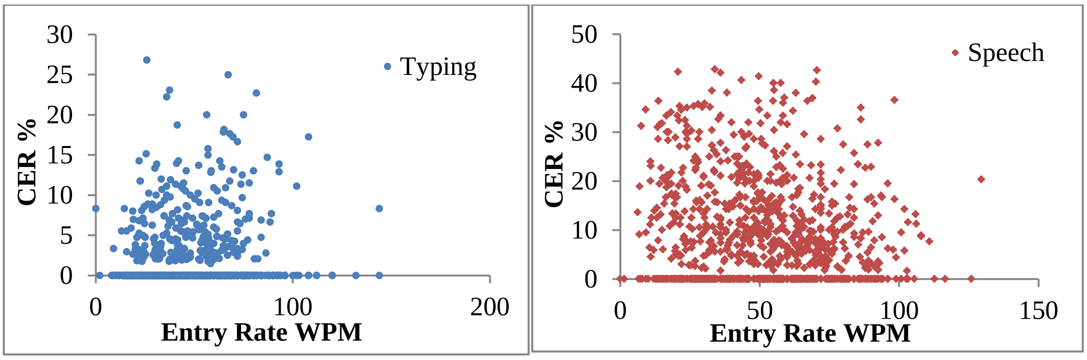


Figure 68 - Entry Rate vs CER Scatter Plots for Typing vs Speech in Experiment D

For analysis purposes we divided these data points based on order into 5 blocks, such that each block contains around 360 data points. Based on this we present the analysis below for Entry and Error Rates.

6.9.2 Entry Rate

We report the following entry rates for this study.

	Mean	Std. Deviation	Minimum	Maximum
Speech	51.31	9.50	37.94	64.30
Typing	47.75	12.11	35.70	63.31

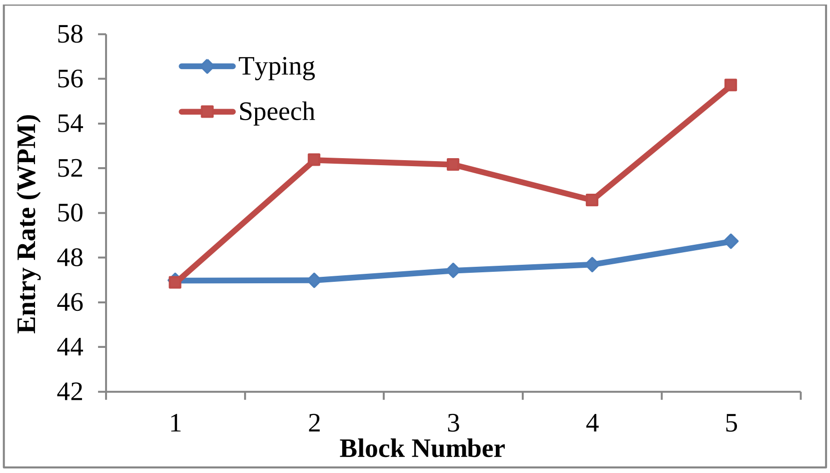


Figure 69 - Entry Rate vs Block for Typing and Speech in Experiment D

Performing ONE-WAY-ANOVA provides the following Between Subjects Effects:

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	38.048	1	38.048	0.321	0.583

This suggests that the difference in entry rate between Typing and Speech in this setting (ESM) is not significant. This was different from the lab study, where speech significantly outperformed typing. We conjecture that this is probably more realistic in terms of entry rate in the real world – as the users were not entirely focussed on the task, were distracted, or in an area with lower network coverage than the lab participants. Which possibly resulted in the lower entry rate for speech. Typing however, doesn't seem to be as affected by these factors.

6.9.3 Character Error Rate

We report the following character error rates for this study.

	Mean	Std. Deviation	Minimum	Maximum
Speech	5.80%	2.39	3.22	10.03
Typing	1.09%	0.97	0.32	2.98

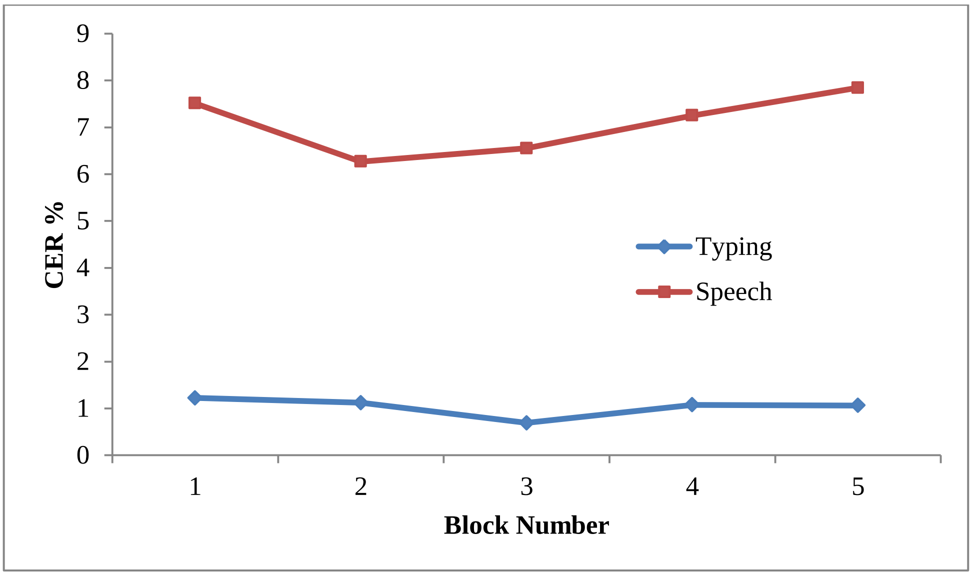


Figure 70 - CER vs Block for Typing and Speech in Experiment D

Performing ONE-WAY-ANOVA provides the following Between Subjects Effects:

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	66.373	1	66.373	19.913	**0.001**

As we can see here, there is a significant difference in the character error rate between speech and typing. We conjecture that this is possibly due to the same factors mentioned overleaf.

6.9.4 Word Error Rate

The Word Error rate follows a close pattern to the Character Error rate, but with higher values for both typing and speech conditions.

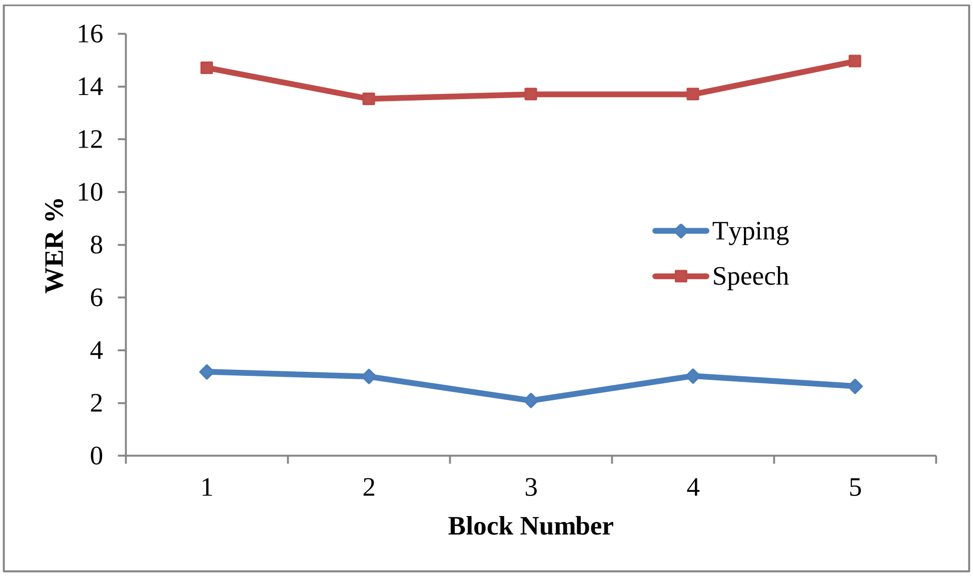


Figure 71 - WER vs Block for Typing and Speech in Experiment D

	Mean	Std. Deviation	Minimum	Maximum
Speech	11.4956	4.11310	6.85	18.33
Typing	2.9925	2.22926	1.07	7.31

We do not present statistical analyses for these as they would follow a similar pattern to CER as shown above.

6.9.5 Evaluating phrases without OOV words

As per the previous studies, we were interested in identifying what happens when OOV words were removed from the study.

6.9.5.1 Entry Rate

It was seen that the entry rate raised slightly in both typing and speech conditions, but didn’t make a difference to the overall result.

	Mean	Std. Deviation	Minimum	Maximum
Speech	51.35	9.63	37.95	64.60
Typing	48.30	12.16	36.00	64.00

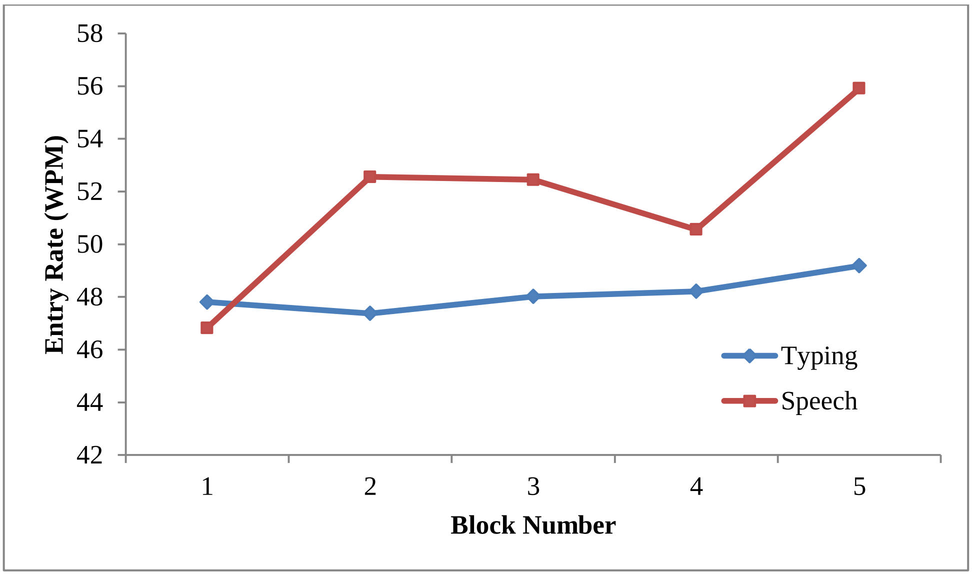


Figure 72 - Entry Rate vs Block for Typing and Speech in Experiment D – without OOV words

Performing ONE-WAY-ANOVA provides the following Between Subjects Effects:

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	27.921	1	27.921	0.232	0.640

As shown here, the difference in entry rate between speech and keyboard was not significant.

6.9.5.2 Error Rate

The error rate followed a similar pattern with the previous comparison for both CER and WER.

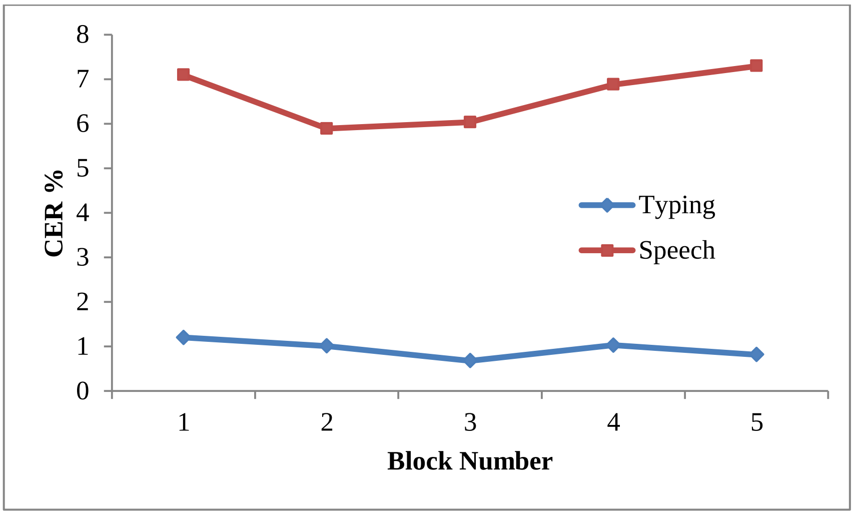


Figure 73 - CER vs Block for Typing and Speech in Experiment D – without OOV words

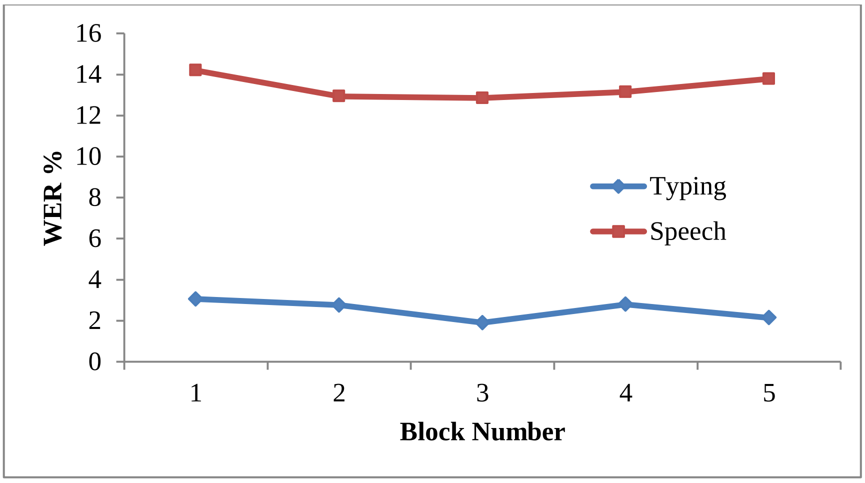


Figure 74 - WER vs Block for Typing and Speech in Experiment D – without OOV words

With the following descriptive statistics:

		Mean	Std. Deviation	Minimum	Maximum
CER	Speech	5.46	2.36	2.97	9.52
	Typing	1.03	0.93	0.33	2.85
WER	Speech	10.92	4.12	6.36	17.52
	Typing	2.76	2.07	1.11	6.85

And the following ONE WAY ANOVA analysis:

	Sum of Squares	df	Mean Square	F	Sig.
CER	58.778	1	58.778	18.240	**0.002**
WER	199.466	1	199.466	18.762	**0.001**

As it can be seen in the plots and as described by the statistics and ONE way ANOVA, there is a significant difference in the CER and WER between speech and typing. Speech clearly produces more errors in the ESM study, which is also similar to the result in the previous (lab based) study.

6.9.6 Phrases with OOV sentences

We were interested in the opposite as well. When considering sentences with contained at least one OOV word in it, we found the following results.

6.9.6.1 Entry Rate

The entry rate for typing and speech both dropped, as shown in the table below.

	Mean	**Std. Deviation**	**Minimum**	**Maximum**
Speech	49.99	6.79	37.84	56.11
Typing	34.28	12.01	21.77	50.48

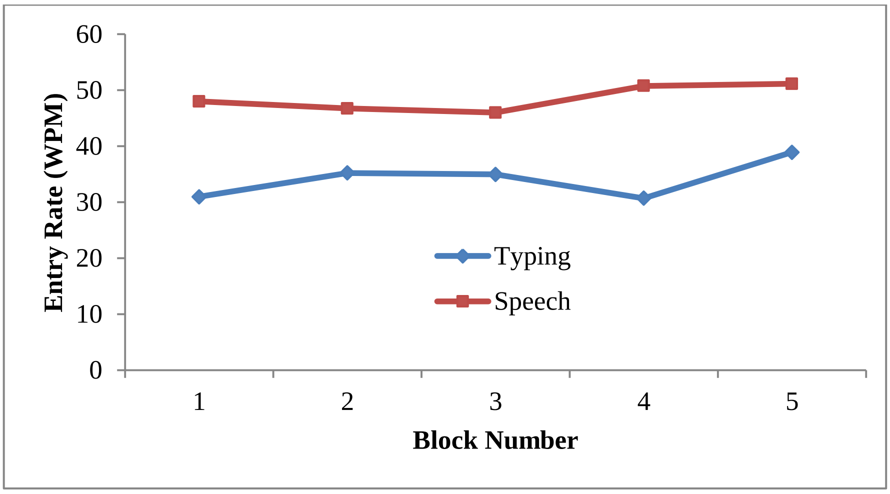


Figure 75 - Entry Rate vs Block for Typing and Speech in Experiment D – considering only OOV phrases

The ONE WAY ANOVA analysis report is as follows:

	Sum of Squares	df	Mean Square	F	Sig.
WPM	740.420	1	740.420	7.784	**0.019**

This shows that the entry rates were significantly different. This suggests that the OOV words have affected typing more in the ESM condition over speech.

6.9.6.2 Error Rate

The character error rate (CER) and word error rate (WER) plots show that with OOV words speech and typing both had a much higher error rate, but with following the same pattern.

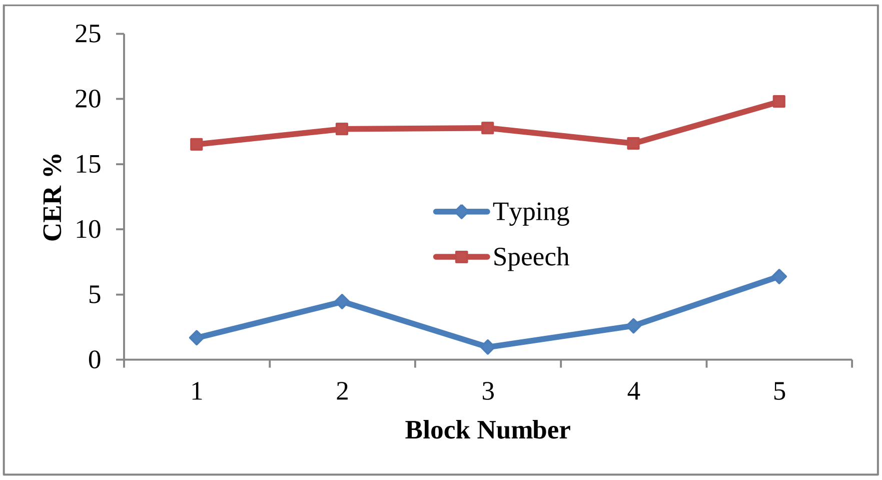


Figure 76 - CER vs Block for Typing and Speech in Experiment D – considering only OOV phrases

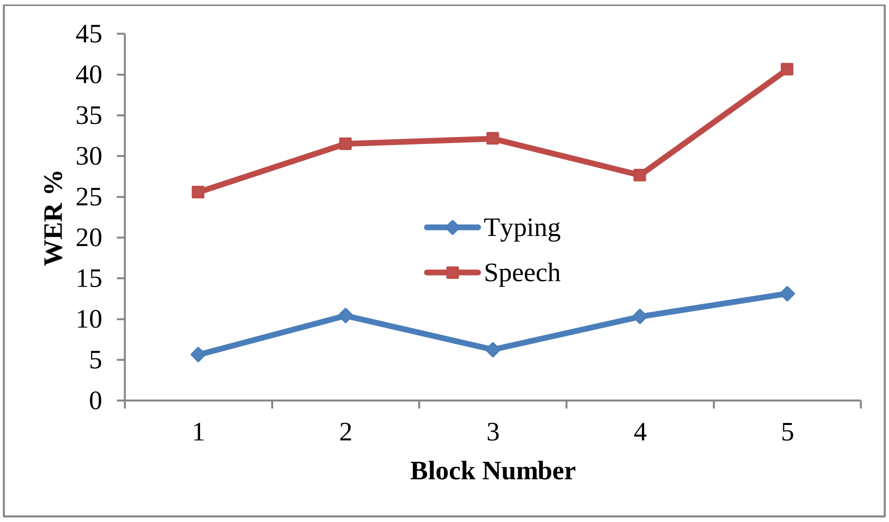


Figure 77 - CER vs Block for Typing and Speech in Experiment D – considering only OOV phrases

The ONE way ANOVA report shows a statistically significant difference in error rates.

	Sum of Squares	df	Mean Square	F	Sig.
CER	393.687	1	393.687	20.634	**0.001**
WER	886.226	1	886.226	11.791	**0.006**

And the following are the descriptive statistics for the CER and WER.

		Mean	Std. Deviation	Minimum	Maximum
CER	Speech	14.11	5.77	7.07	22.36
	Typing	2.66	2.22	0.00	6.18
WER	Speech	25.70	10.24	11.11	38.04
	Typing	8.51	6.74	0.00	19.07

6.9.7 Perplexity Analysis

As explained in the Materials section under Phrase Set, we divided the phrase set into quartiles based on perplexity. Based on this, we analysed how the entry rates and error rates would differ between typing and speech with varying perplexity.

6.9.7.1 Entry Rate

When perplexity increases, the entry rate dropped slightly for both the input mechanisms, however it can be seen that typing was more effected than speech in the ESM study.

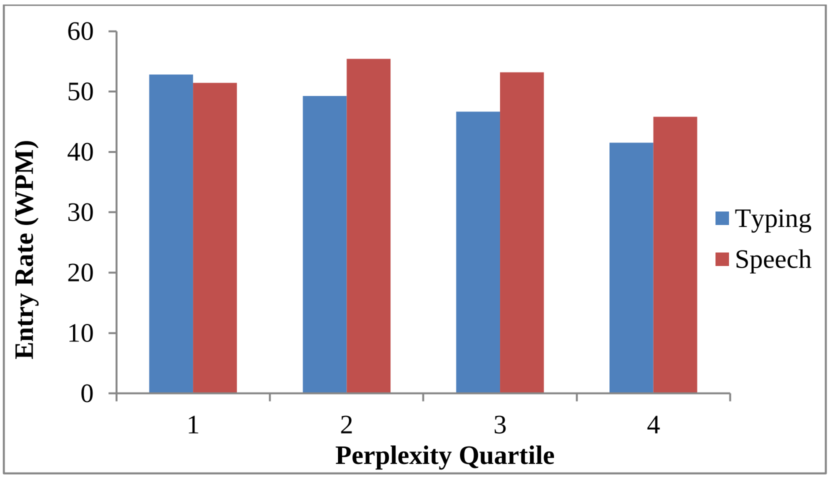


Figure 78 - Entry Rate vs Perplexity for Typing and Speech in Experiment D

WPM	Typing		Speech	
	Mean	SD	Mean	SD
Quartile 1	53.50	11.86	50.5	10.6
Quartile 2	49.51	11.99	55.0	9.2
Quartile 3	46.68	13.55	53.4	10.0
Quartile 4	41.37	11.65	46.2	9.1

WPM	F_{df}	df	η_p^2	p
Input Method	.285	1	.054	.616
Quartile	31.134	3	.862	**.000**
IM x Q	14.295	3	.741	**.000**

The results and analysis clearly show that perplexity has a significant effect on entry rate for both conditions (typing and keyboard), and keyboard has been affected more than speech.

6.9.7.2 Error Rates

This is possibly the most interesting result of all, which sheds light on the ability of speech recognition to deal with complex sentences. When the perplexity increased, the error rate (both CER and WER) raised geometrically for speech, whereas the error rate for typing raised at a much lesser rate.

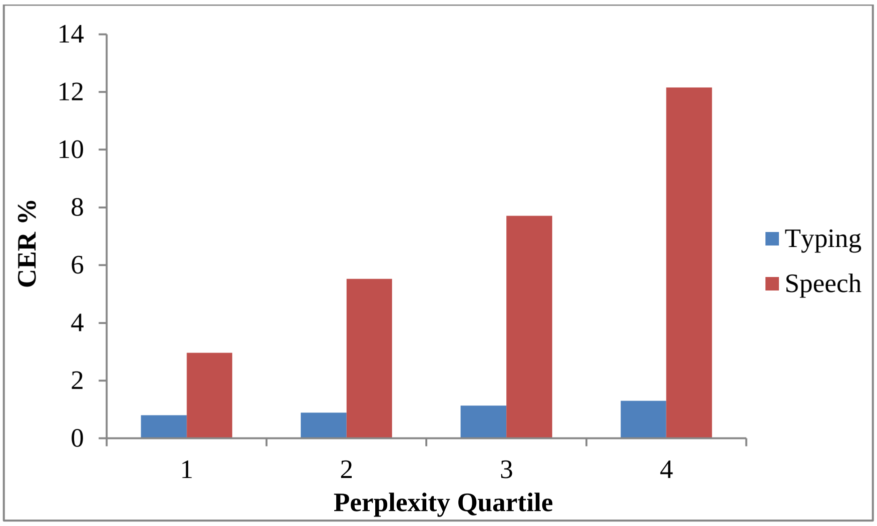


Figure 79 - CER vs Perplexity for Typing and Speech in Experiment D

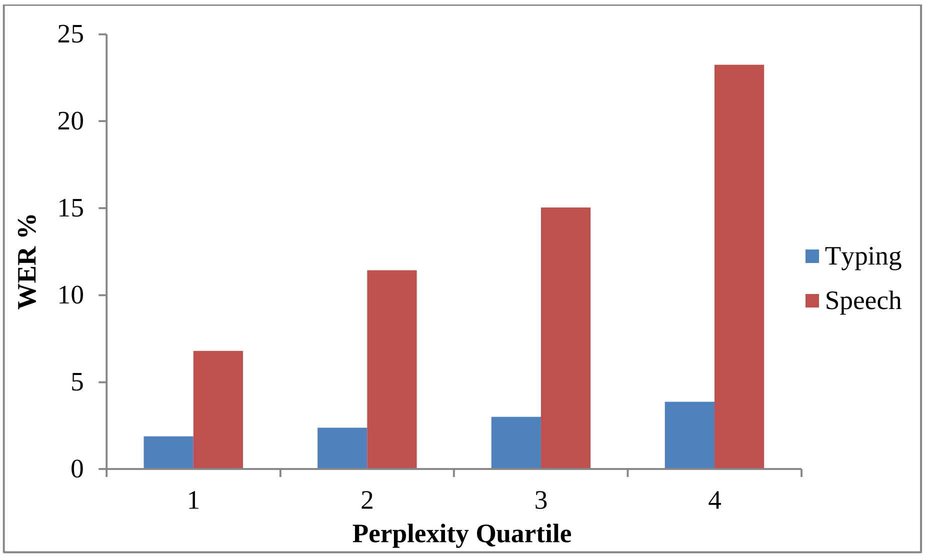


Figure 80 - WER vs Perplexity for Typing and Speech in Experiment D

CER%	F_{df}	df	η_p^2	p
Input Method	19.724	1	.798	**.007**
Quartile	23.998	3	.828	**.000**
IM x Q	17.683	3	.780	**.008**

Noting the significant interaction between Input Method x Quartile - this clearly states that speech was affected more by perplexity than typing during this study.

The descriptive statistics for the above CER% are as follows:

CER%	Typing		Speech	
	Mean	SD	Mean	SD
Quartile 1	0.91	0.64	2.26	1.68
Quartile 2	1.08	1.32	4.86	1.69
Quartile 3	1.20	1.23	6.17	3.69
Quartile 4	1.17	0.80	9.94	3.51

6.9.8 Subjective Ratings

Since this was analysed as a between subjects study, we did not collect subjective ratings comparing the two input methods as it would be meaningless. Once all the participants experience both treatments, then we shall collect, analyse and publish these results.

6.9.9 Open Comments

We however asked the users to provide open comments on what they liked and disliked about each input method, and what would be their decision making factors when choosing one over the other.

6.9.9.1 Like about Keyboard

1. “Quicker and more intuitive”
2. “Faster and accurate than voice”
3. “It can be done in any condition and any context.”
4. “you can send messages when you would not normally be able to speak.”
5. “Doesn't disturb others”
6. “Familiar method as I have been using it all this time”
7. “It tends to work quite fast for most simpl simple sentences.”
8. “Easy to use”

6.9.9.2 Dislike about Keyboard

9. “Frequent typing errors - big fingers”
10. “It is slow, cumbersome when on the go, lots of mistakes, especially when trying to type in languages other than English.”
11. “Can be frustrating when errors are encountered”
12. “typing a very long sentence or few sentences it's annoying,”
13. “have to keep attention to the screen”

6.9.9.3 Like about Speech

14. “Practical method if you can't type”
15. “It is fast, one of the most convenient ways of communicating on the go.”
16. “Fast”
17. “it's easy, fast and it doesn't require too much effort”
18. “Also no need to keep eyes peeled to the screen”
19. “It's very straight forward”

6.9.9.4 Dislike about Speech

20. “Difficulties in recognizing the words”
21. “Not as fast or accurate as keyboard for me. Is not accurate for all accents.”
22. “You would not be able to use it anywhere because of the noise disruption it causes to others around you.”
23. “Disturbs other people, feel awkward doing it in a public space”
24. “Can be inaccurate, depending on accent.”
25. “it's inconvenient when you are in public spaces”
26. “The error rate can be high depending on the complexity of sentence and pronunciation of some words, “
27. “cant use this method in a crowded spot”
28. “When it doesn't work, I need to re say everything.”
29. “Cannot recognise some words”

6.9.9.5 When will you choose Keyboard over Speech

30. “Environments those are too quiet to use speech”
31. “I could be using it anywhere without disturbing people.”
32. “Driving, or at home with nobody around”
33. “if I'm in a crowded area or meeting, or typing a complex sentence”
34. “Privacy and not disturbing others.”

6.9.9.6 When will you choosing Speech over Keyboard

35. “If I am not able to type at the moment”
36. “If I had problems with my sight, I would avail of this option.”
37. “When I have a longer message to send.”
38. “If I am in the car and need to send a message while waiting at the lights (safely)”
39. “Multitasking (i.e. cooking and texting someone)”
40. “when I need to do a task very fast and I can't use both of my hand at the moment”
41. “Driving”
42. “If I'm alone in a quiet environment and i feel lazy.”

6.9.9.7 Analysis of open comments

We divided the comments into the follow categories

	Typing		Speech	
	Positive	**Negative**	**Positive**	**Negative**
Speed	1,2,7	10	15,16,17	
General Accuracy	2,7	9,11		20,21,24,29
Convenience	1,3,4,6,8	12,13	14,15,17,18,19	
Editing/Correction				28
OOV, Perplexity, Punctuation				26
Privacy & Social	5			22,23,25,27

These reviews were slightly more mixed compared the lab study (chapter 5). However, we can see a general trend in the positive and negative aspects of each input method following the same pattern as before. Speech was preferred for its speed and convenience e.g. typing without the hands, or driving etc. and was not preferred for its social and privacy aspects. Further accuracy, with or without OOV, and editing/correction were also concerns. When examining the open comments on why would one choose keyboard over speech, or vice versa, the comments suggest that users would choose speech if the conditions were fitting – e.g. not crowded, not disturbing others, or limitations such as not being able to use their hands or eyes i.e. visually impaired, or driving, or multitasking. On the other hand, when accuracy, precision and privacy are priority, then keyboard would be chosen.

6.10 Summary

The following is a summary of the quantitative analysis performed on the results of this study.

6.10.1 Entry Rates

In this table, we report the entry rates from the various analyses performed.

Mean Entry Rates	Normal	Excluding OOV Words	Only Sentences with OOV Words
Typing	47.7 (SD=12.1)	48.3 (SD=12.2)	34.28 (SD=12.01)
Speech	51.3 (SD=9.5)	51.4 (SD=9.6)	49.9 (SD=6.8)
Is difference significant?	No	No	Yes

So this tells us that, in the wild, under a normal phrase set mixed between OOV and non-OOV words, speech slightly outperforms typing in terms of entry rate, but not significantly. However, when entering sentences with at least OOV word, the entry rate for typing drops much lower than of speech, therefore having significant difference in the entry rates then. This indicates that the presence of OOV's have affected typing more than speech, in the wild.

Therefore we do not have enough evidence to reject the following null hypotheses:

H0,a There is no difference in text entry rate between the keyboard and speech in the wild

However, we can reject this null hypothesis

H0,c The presence of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of entry rate, in the wild

And replace it with this alternate hypothesis:

H1,c The presence of Out of Vocabulary (OOV) affect keyboard more than speech in terms of entry rate, in the wild

6.10.2 Error Rates

In this table, we will consider only Character Error Rates (CER) as this was the basis for the statistical analysis. Word Error Rates were observed to follow a similar pattern with higher values.

Mean Error Rates	Normal	Excluding OOV Words	Only Sentences with OOV Words
Typing	1.09% (SD=0.97)	1.03% (SD=0.93)	2.66% (SD=2.22)
Speech	5.80% (SD=2.39)	5.46% (SD=2.36)	14.11% (SD-5.77)
Is difference significant?	Yes	Yes	Yes

So this tells us that, in the wild, under a normal phrase set mixed between OOV and non-OOV words, typing outperforms speech in terms of error rate.

Therefore we have enough evidence to reject the following null hypothesis:

H0,b There is no difference in character error rate after correction between keyboard and speech in the wild

And replace them with the following alternate hypothesis:

H1,b Speech produces more errors than typing in the wild

However, we do not have enough evidence to reject the following null hypothesis, as with and without OOV words, there is still a significant difference in error rates between the two input methods.

H0,d The presence of of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of error rate , in the wild

6.10.3 Sentence Perplexity

	Significant Effect of Input Method	Significant Effect of Quartile	Interaction between IM x Q
Entry Rate	No	Yes	Yes
Error Rate	Yes	Yes	Yes

This table, in combination with the above results, clearly tells us that when perplexity varies, it affected one input method more than the other. When comparing the values, we can see that speech is affected at a far higher level than keyboard in terms of error rate, and keyboard is affected in terms of entry rate.

Therefore, we can reject the following null hypotheses:

H0,g The perplexity of the phrase does not affect keyboard and speech differently in terms of entry rate, in the wild

H0,h The perplexity of the phrase does not affect keyboard and speech different in terms of error rate, in the wild

And replace it with these alternate hypotheses:

H1,g The perplexity of the phrase affects keyboard more than speech in terms of entry rate, in the wild

H1,h The perplexity of the phrase affects speech more than keyboard in terms of error rate, in the wild

6.10.4 Subjective Ratings & Open Comments

The open comments provide an indication of as to what aspects users liked and disliked about each input mechanism. However, since we were unable to analyse the results from subjective ratings, we do not have enough evidence to reject the following null hypotheses:

H0,e The user experience of the participants did not differ between keyboard and speech

H0,f In the lab, users did not prefer to use one method over the other when given a choice between both

6.11 Conclusions

All in all, we could draw the following conclusions from this study:

H0,a There is no difference in text entry rate between the keyboard and speech in the wild

H1,b Speech produces more errors than typing in the wild

H1,c The presence of Out of Vocabulary (OOV) affect keyboard more than speech in terms of entry rate, in the wild

H0,d The presence of of Out of Vocabulary (OOV) words do not affect keyboard and speech differently in terms of error rate , in the wild
-- Not enough evidence to say otherwise

H0,e The user experience of the participants did not differ between keyboard and speech in the wild
-- Not enough evidence to say otherwise

H0,f Users did not prefer to use one method over the other when given a choice between both, in the wild
-- Not enough evidence to say otherwise

H1,g The perplexity of the phrase affects keyboard more than speech in terms of entry rate, in the wild

H1,h The perplexity of the phrase affects speech more than keyboard in terms of error rate, in the wild

7

Discussion

In this chapter, we provide a discussion on the results from chapters 3 to 6.

7.1 Summary – STK vs. SGK

In this section, we present and reflect on the findings from chapters 3 and 4.

7.1.1 Entry Rate

The entry rate results across all four studies are as follows. All results are shown in Words per Minute (WPM) – in the format of mean followed by standard deviation within brackets.

7.1.1.1 Normal – Full dataset

Lab	STK	34.66 (10.2)	STK was significantly faster
	SGK	32.02 (9.64)	
ESM	STK	33.9 (11.8)	SGK was significantly faster
	SGK	40.5 (15.5)	

7.1.1.2 Dataset excluding OOV words

Lab	STK	34.89 (10.2)	STK was significantly faster
	SGK	32.37 (9.5)	
ESM	STK	34.1 (11.9)	SGK was significantly faster
	SGK	40.9 (15.5)	

7.1.1.3 Dataset with only phrases including OOV words

Lab	STK	29.55 (8.8)	STK was significantly faster --- ***Presence of OOVs affected SGK more***
	SGK	23.85 (9.33)	
ESM	STK	31.4 (13.7)	No significant difference in speed for STK and SGK --- ***Presence of OOVs affected SGK more***
	SGK	30.5 (9.6)	

7.1.1.4 Effect of Hand Postures

Lab	STK	Two Thumb	35.75 (9.99	Fastest: --- Two-thumb STK --- Single Finger SGK
		Single Finger	25.69 (9.85)	
		Single Thumb	26.63 (7.68)	
	SGK	Two Thumb	-- not used	
		Single Finger	34.75 (9.85	
		Single Thumb	30.11 (9.01)	
ESM	STK	Two Thumb	36.02 (11.5)	Fastest: --- Single Thumb SGK ---Single Finger SGK --- Two Thumb STK
		Single Finger	-- not used	
		Single Thumb	29.77 (11.39)	
	SGK	Two Thumb	-- not used	
		Single Finger	38.59 (13)	
		Single Thumb	41.00 (16.1)	

7.1.2 Error Rate

Results shown are CER % in the format of mean, followed by standard deviation in brackets.

7.1.2.1 Normal – Full Dataset

Lab	STK	0.91 (2.57)	SGK produced significantly more errors
	SGK	2.05 (4.27)	
ESM	STK	1.76 (3.88)	
	SGK	3.36 (6.04)	

7.1.2.2 Dataset excluding OOV words

Lab	STK	0.89 (2.55)	SGK produced significantly more errors
	SGK	1.86 (4.14)	
ESM	STK	1.72 (3.87)	
	SGK	3.18 (5.95)	

7.1.2.3 Dataset with only phases including OOV words

Lab	STK	1.29 (2.86)	SGK produced significantly more errors
	SGK	5.49 (5.60)	
ESM	STK	2.59 (4.04)	
	SGK	7.55 (6.98)	

7.1.2.4 Effect of Hand Postures

Lab	STK	Two Thumb	0.75 (2.82)	Least errors: --- Two thumb STK --- Single Thumb STK --- Single Thumb SGK Interesting: --- Bi-Manual SGK not used
		Single Finger	3.96 (1.87)	
		Single Thumb	0.79 (2.33)	
	SGK	Two Thumb	-- not used	
		Single Finger	3.06 (5.15)	
		Single Thumb	1.03 (3.34)	
ESM	STK	Two Thumb	1.43 (3.42)	Least errors: --- Two thumb STK --- Single Thumb STK --- Single Finger SGK
		Single Finger	-- not used	
		Single Thumb	2.43 (4.59)	
	SGK	Two Thumb	-- not used	

		Single Finger	2.46 (4.93)	Interesting: --- Bi-Manual SGK and single finger STK not used
		Single Thumb	3.63 (6.31)	

7.1.3 User Ratings

Lab	STK preferred	Rated Higher than SGK for Speed, Ease of Use, Preference Not Rated Higher than SGK for Accuracy
ESM	SGK preferred	Displayed a clear migration of users from STK to SGK

7.1.4 Open Comments

Lab	STK	Preferred for control, error correction and ability to enter OOVs
	SGK	Preferred for general user experience
ESM	STK	Preferred for control, error correction, and ability to enter OOVs
	SGK	Users were willing to adopt SGK, and preferred for ease of use

7.2 Reflections – STK vs. SGK

It is interesting to see that the two methods yielded such different results, despite the two keyboards used being the same. We understand that two studies, designed in completely different ways, with different participants are bound to yield different results, therefore we do not heavily cross compare across the two methodologies to decide which one is better. We simply yield the results from both studies to widen our understanding on mobile text input and conclude that both types of experiments are required when empirically comparing text input methods.

7.2.1.1 Does STK perform faster in controlled environments?

In the lab study we observed that overall STK was significantly faster in our sample. This can be understood by studying the participants' experience. Only five out of twelve participants were Android users, and out of these, only three were familiar with SGK. Therefore, participants had previous experience with STK, whereas SGK was a completely new experience for them. Also, the mean entry rate difference between SGK and STK was greater in the beginning of the study, and the difference decreased towards the end, as participants significantly improved with practice on both conditions. This indicates that with practice, users become faster with SGK as they continue to use it.

In the ESM study, the overall entry rate for SGK was significantly higher compared to STK. There are two plausible reasons for this discrepancy across the studies. One is that the participants in the ESM study were more experienced with SGK than in the lab study, and also they were using their own mobile devices, which they were already familiar with.

We conjecture that the second reason might the environment. In the lab study, participants were seated, fully focused on the task, and not distracted by any other simultaneous task. This situation would be ideal for STK, where users can reach staggering speeds of up to 84 WPM. This is not the case in the ESM study, where most of the time participants were not expecting a task (as they were requested randomly during the day), and therefore the participants were most likely busy with other activities, such as being on the move. According to the figure below (from Chapter 4), It was clear that the participants were mostly on the move, distracted, or engaged in some other task when they were requested to participate in the study.

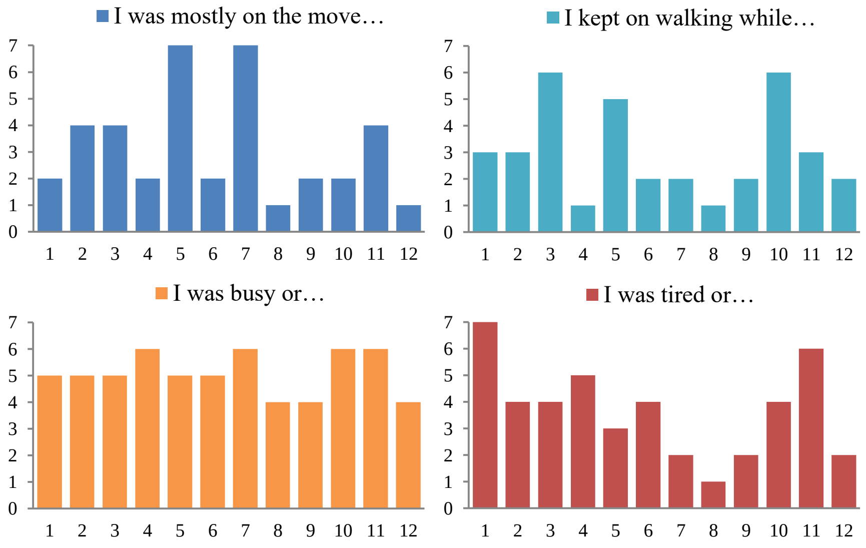


Figure 81 - Subjective ratings on movement and distraction by participants in Experiment B

Taking into account all these and other external factors (such as distraction, exhaustion, movement) it might be easier for the participants to slide their fingers across the touchscreen by making recognizable gestures from motor memory rather than needing to rely on visually-guided tapping motions on small targets. In the ESM study, we found that some participants reached very fast entry rates for SGK, up to 120 WPM, while transcribing common phrases such as “weather is bad here”.

7.2.1.2 Why does SGK produce higher error rates?

Comparing the error rates between the two experiments, SGK results in significantly higher error rates, whether it being CER% or WER%. Participants’ open comments provided for each input method included remarks such as stating instances where they could enter non-dictionary words in STK, while using the SGK the option wasn’t available. This problem of Out of Vocabulary (OOV) errors is shared among other intelligent text entry methods, such as speech recognition – see the reflection on Chapters 5 and 6.

As we identified, our subset of Enron mobile phrase set contained around 44 (3%) non-dictionary words, such as names and places. SGK users could not enter these words, whereas STK users could always ignore the autocorrected suggestion and revert back to the original word based on their key tap positions. This option is not available in SGK. Therefore, SGK inevitably produced incorrect words with no way for the participant to correct them without switching into STK.

7.2.1.3 Investigating data with no OOV words

As mentioned in the results section we investigated OOV’s in more depth in a series of post-hoc analyses. We noted that excluding OOV’s had a greater impact on the entry rate of SGK than STK, as in the lab study, the difference between entry rates were no longer significant. Also it was seen that the grand mean entry rate rose higher for SGK in the ESM study when OOV’s were excluded. But this was marginal, and in both experiments SGK still produced a higher error rate.

7.2.1.4 Investigating data with at least one OOV word in a sentence

Our analyses of just the data points containing OOV’s (the excluded points above) were able to shed more light on the matter. Now in both the studies, the entry rate was almost similar between the conditions. In the ESM study, SGK was previously significantly faster than STK, but the rate dropped sharply when we considered only the OOV sentences. This means OOV’s impacted SGK entry rates more than STK. Also for the OOV words both STK and SGK produced a very high error rate (CER and WER), thus we also need to accept that STK is also affected substantially by OOV’s but not as much as SGK.

How OOV’s affect SGK entry rate can be explained. When using SGK, if the user doesn’t get the word he/she expected then it could be for two reasons. It is either the user got a position or part of the gesture incorrect or the word is not in the lexicon. Normally at the first instance the user believes it is the prior, and deletes the word (a single backspace deletes the entire work on SGK) and tries again. After a few tries, perhaps the latter tries being slower and more accurate than the first, the user finally realizes this is an OOV and then either switches to STK to complete the word or simply ignores the incorrect words and completes the rest of the phrase. The clock keeps running during this trial and error process. Our participants took either of the above two decisions when entering text, so either ended up with a small entry rate or a greater error rate. STK isn’t affected so much by this because if an STK user gets a word wrong usually he/she can either select the original word, ignoring the corrected word or simply fix one or two characters and get the correct word, which takes substantially less time.

7.2.1.5 Investigating Hand Posture

It should be noted that we didn’t control hand posture, but rather asked the users to go with their most proffered, but make a note of it. Even though this was the case, the number of users across the two studies who used the same hand posture are comparable (i.e. 2-thumb STK was used by 7 users in the lab study and 8 users in the ESM study).

In STK, 2-thumb was by far the best, as it produced the highest input rates and smallest error rates in both studies. This makes input methods such as KALQ better for touchscreen input as it relies on this hand posture. Also in the lab study, 2-thumb STK rated better than all other STK and SGK hand postures in terms of speed, error rate and user rating, thus making it the best variant for such environments.

In the SGK condition, single finger produced the fastest entry rate and smallest error rate in the lab study. In the ESM study, both SGK hand postures were faster than STK ones. Single thumb was the fastest in SGK, with the smallest error rate, and 2-thumb was the fastest in STK. This contrasts with the lab study as single finger SGK was the fastest and smallest error rate.

Single finger SGK and single-finger STK use both hands, where the user holds the device in the non-dominant hand the uses the index finger of the dominant hand to tap/gesture. This requires the coordination of two independent hands. Single thumb STK and single thumb SGK are single handed operations, where the user holds the device in the same hand uses the thumb to tap/gesture. 2-thumb STK and bi-manual SGK are unique as they use both hands, but both hands act as one, constraining each other's movements by holding the phone in both hands, and using both thumbs (possibly alternatively) to tap/gesture. Since the movements of both hands are constrained the relative movements between the hands are minimal.

In the lab study the users were seated, had full visual feedback and had their elbow(s) rested on a table despite which hand posture they used. In the ESM study we speculate their arms were freely moving, and the subjects were possibly standing, on the move, and perhaps relying on motor memory. In the latter, having a more constraint hand posture (with less relative motion between the hands) such as single thumb, or 2-thumb would yield faster and accurate text entry than single finger. This would explain why in SGK, single finger was faster and more accurate in the lab, whilst single thumb was faster and more accurate in the wild.

7.2.1.6 User preferences

Another interesting outcome from these two studies was the qualitative results, which we collected from the participants in the form of subjective ratings and open comments. Importantly our studies also revealed how users' preferences changed over time. We found that the participants demonstrated a definite trend towards moving to SGK, a still novel method to many mobile users (as is illustrated below – copied from Chapter 4); to paraphrase one of our participant's feedback: this experiment may have changed a few lives.

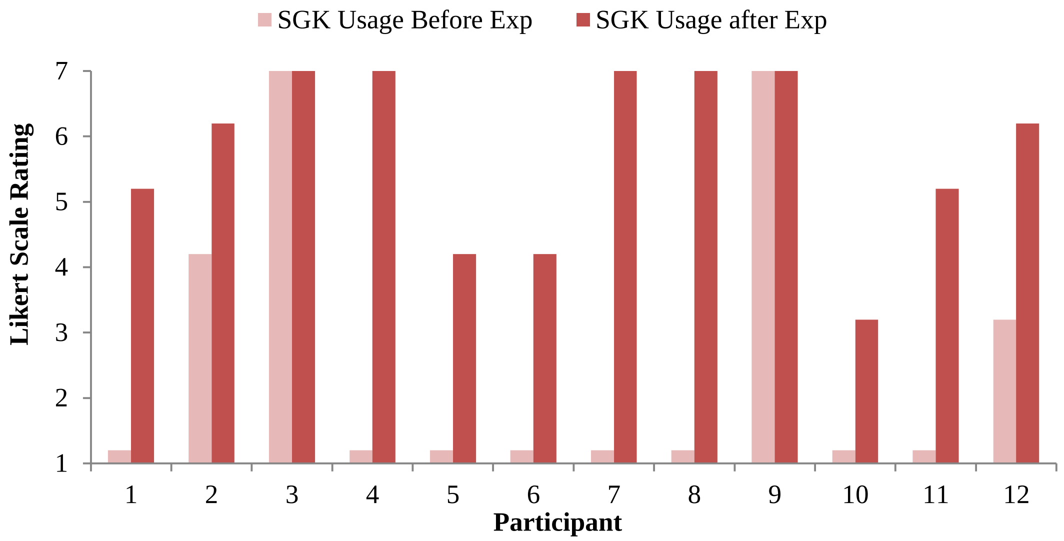


Figure 82 - The adoption of SGK in Experiment B

The open comments by the majority of the participants revealed that they found the SGK to be a more natural, enjoyable, fast and accurate experience, especially when being on the move. In the lab study, we observed that participants' open comments mostly focused on accuracy and speed, and in the lab-study participants stated they found the STK to be faster, more accurate and easier, but in the ESM study, a different set of participants found SGK to be more desirable. This raises questions about the ecological validity of the lab study-based evaluation paradigm that is used in nearly all text entry studies.

7.3 Summary – Typing vs. Speech

In this section, we present and reflect on the findings from chapters 5 and 6.

7.3.1 Entry Rate

The entry rate results across all four studies are as follows. All results are shown in Words per Minute (WPM) – in the format of mean followed by standard deviation within brackets.

7.3.1.1 Normal – Full dataset

Lab	Typing	40.23 (12.27)	Speech was significantly faster
	Speech	58.88 (21.51)	
ESM	Typing	47.7 (12.1)	Entry rate difference not significant
	Speech	51.3 (9.5)	

7.3.1.2 Dataset excluding OOV words

Lab	Typing	40.60 (12.27)	Speech was significantly faster
	Speech	59.22 (21.38)	
ESM	Typing	48.3 (12.2)	Entry rate difference not significant
	Speech	51.4 (9.6)	

7.3.1.3 Dataset with only phrases including OOV words

Lab	Typing	31.44 (8.51)	Speech was significantly faster
	Speech	50.75 (22.83)	
ESM	Typing	34.28 (12.01)	Speech was significantly faster --- ***Presence of OOV's affected TYPING more***
	Speech	49.9 (6.8)	

7.3.1.4 Effect of Perplexity

Lab	Speech had a higher entry rate Entry rate dropped across quartiles 1 → 4 ***SPEECH was affected more by perplexity***

ESM	No significant difference in entry rate between typing and speech Error rate increased across quartiles 1 → 4 ***TYPING was affected more by perplexity***

7.3.2 Error Rate

The entry rate results across all four studies are as follows. All results shown are Character Error Rates (CER) in as % in the format of mean, followed by standard deviation in brackets.

7.3.2.1 Normal – Full Dataset

Lab	Typing	0.22 (0.99)	Speech produced significantly more errors
	Speech	3.34 (6.84)	
ESM	Typing	1.09 (0.97)	
	Speech	5.80 (2.39)	

7.3.2.2 Dataset excluding OOV words

Lab	Typing	0.20 (0.94)	Speech produced significantly more errors
	Speech	3.07 (6.56)	
ESM	Typing	1.03 (0.93)	
	Speech	5.46 (2.36)	

7.3.2.3 Dataset with only phases including OOV words

Lab	Typing	0.70 (1.69)	Speech produced significantly more errors
	Speech	10.09 (9.54)	
ESM	Typing	2.66 (2.22)	
	Speech	14.11 (5.77)	

7.3.2.4 Effect of Perplexity

Lab	Speech had a higher error rate Error rate increased across quartiles 1 → 4 ***SPEECH was affected more by perplexity***
ESM	

7.3.3 User Ratings

The user ratings are in combination for entry rate, error rate and user experience.

Lab	Typing	No significant difference in perceived input speed, accuracy, ease of use or preference
	Speech	
ESM	Typing	*Not presented as between subjects analysis*
	Speech	

7.3.4 Open Comments

The user ratings are in combination for entry rate, error rate and user experience.

Lab ESM	Typing	Preferred for accuracy, ability to enter OOVs, control and editing/correction, and privacy/social factors – disliked for inconvenience, hogging attention, focus and motor freedom
	Speech	Preferred for speed, and convenience – disliked for accuracy, error correction, control over what is typed, and social/privacy factors

7.4 Reflections – Typing vs. Speech

Similar to the STK vs. SGK study, it was very interesting to see the two methods two methods yield different results. Again, we understand that two studies, designed in completely different ways, with different participants are bound to yield different results, therefore we do not heavily cross compare across the two methodologies to decide which one is better. We simply yield the results from both studies to widen our understanding on mobile text input and speech input to conclude that both types of experiments are required when empirically comparing text entry mechanisms.

7.4.1.1 Does Speech perform faster in controlled environments?

This was a clear observation when analysing the results of both the studies. In the lab study, speech clearly outperformed typing in terms of entry rate (59 WPM vs 40 WPM). The highest data points being around 127.2 WPM – for one particular user speaking “'I thought we already reached an agreement to buy them” in under 5 seconds with a 0% error. In comparison, the highest speed recorded for typing was around 90 WPM – where a user typed “He doesn't want to give the trading positions” in little over 6 seconds with a 0% error rate.

Human beings can speak up to 200 WPM (Rosenbaum, 1991) and of course when incentivized during the study to perform as fast and accurately as possible, would perform at their best. This is complemented by the study environment provided – a quiet background, full focus on the task, full visual feedback, a brand new mobile device with no other apps running, and the full backing of a super-speed W-Fi connection. However, the average or highest speeds for speech input are nowhere close to this value. We observed that this is because it does take finite amount of time to start the speech recognizer – a slight delay when the mic button is pressed and it starts listening, and a slight delay to decode the sentence and render it after the user stops speaking. This of course should be (and is) factored into the entry rate which gives us the realistic upper bound of speech input on the current state of the art in a controlled environment.

However, this was clearly not the case in the ESM study. We observe that when considering the full dataset and subset with no OOV words, speech input did not perform significantly faster than typing. The average entry rates for speech and typing were around 51 and 48 WPM respectively. The fastest entry rate recorded for speech in the ESM study with a 0% CER was 126WPM, for transcribing “this seems fine to me” in little over 2 seconds. The fastest typing speed recorded in the ESM study was around 140WPM transcribing “are you sure” in slightly over 1 second with a 0% error rate. This performance is possible with a very high level of SGK experience or a very effective prediction system where gesturing “are” and then doing two taps on the suggestion bar resulting in “you” and “sure” consecutively, followed immediately by pressing NEXT.

Another interesting observation is that, when comparing entry rates across the two studies, typing performed faster in the ESM study.

We find two plausible reasons for why typing was faster in the ESM study. The first is that participants in the ESM study were using their own mobile devices, which they were already familiar with. Most of the participants in the Lab study were Apple users, who were used to the iOS keyboard, which had significantly different user experience than Google Keyboard on the Huawei P20 Lite Android device we provided. The second reason is that, in the lab study, all the participants used two-thumb STK to enter text on the typing condition – most continuing their usual practice being iPhone users. However, in the ESM study, several users were familiar with SGK – 7 out of 12 participants rated themselves having above average experience and performance on it. This agrees with the results from the previous ESM study (STK vs SGK) which shows as that SGK can perform faster than STK in the wild. This would explain the faster typing speeds in the ESM study.

We also find several plausible reasons for why speech was slower in the ESM study than the lab study. As this was conducted “in the wild”, the participants would have to transcribe text using speech in environments with high levels of background noise, and low levels of network connectivity – both of which impacts speech recognition adversely. Upon interviewing the participants we noted that, in both these “hostile” environments, the delay in decoding a sentence after the user has finished speaking is longer. Further, the participants mentioned that in very hostile conditions, the decoded text is completely different from what was intended; therefore they had to delete everything and start again. This of course, would impact the entry rate as clearly shown in the results.

7.4.1.2 Does speech produce more errors?

Another interesting observation was that in both the experiments, speech produced a higher error rate than typing. This can be directly attributed to the users open comments about speech recognition not being able to handle errors well – i.e. a user had no way of going back and correcting it without switching to typing. Upon interviewing the users, we found that users tried to correct the errors by going back and verbally repeating the misspelt word a few times, and if this did not work, they simply chose to continue to the next sentence. This also coincides with the same problem SGK had also suffered in the earlier ESM study.

Also we noted that the error rates were higher in the ESM study (5.80%) as opposed to the lab study (3.34%). This can be attributed to the same "hostile" conditions mentioned overleaf – high levels of background noise and low levels of network connectivity, which impacted the entry rate of speech in the ESM study.

7.4.1.3 Perplexity and the presence of OOVs

As described in the introductions chapter, Perplexity is the measure of complexity of a sentence, directly attributed to the branching factor of how many possibilities exist for the next word, given what is already entered. And we observed that the perplexity and presence of OOV's have a correlation – the higher perplexity quartiles contained more sentences with OOV words in them, as shown here:

- 1st Quartile – 0 sentences with OOV words
- 2nd Quartile – 1 sentence with OOV words
- 3rd Quartile – 2 sentences with OOV words
- 4th Quartile – 24 sentences with OOV words

This would mean that, for reasons mentioned above, it would be more difficult to transcribe a sentence with higher perplexity, and would be even more difficult with speech. This aligns with the results from both the studies – where the error rate significantly increased with perplexity, but the error rate of speech rose at a faster rate than of typing.

With regards to entry rate, the effect of perplexity and the presence of OOV's were slightly more complex. In the lab study, the perplexity and entry rate patterns for speech followed a similar trend to that of error rate – i.e. more perplexity, worse performance (higher error rate), and lower entry rate. This is clear as the statistics show in the lab study, perplexity affects speech entry rate more than typing. This can be attributed to the reasons mentioned earlier in this section.

However, in the ESM study, it was noted that when considering the dataset that had at least one OOV per sentence, speech was observed to be significantly faster than typing – where as in the normal and no-OOV datasets, there was no significant difference in the entry rates. This means that the presence of OOV's in the lab study has affected typing more. Similarly, the analysis on the perplexity vs entry rate shows that in the lab study, perplexity has affected typing more than speech. The reason for this can be attributed to the fact that most of the users in the ESM study were SGK users, and SGK suffers from the same issues that speech does such as not being able to predict OOV words and not being able to correct without having to switch to STK (in this case, typing). When interviewing the participants, it was noted that most of them had used SGK to type; however, upon incurring an incorrectly guessed word, they had gone back and corrected the word using STK – as this study specified the condition as simply "typing" and not locking in a participant to either STK or SGK. This of course, had affected the typing duration and therefore the entry rate, but kept the error rate low.

7.4.1.4 The adoption of speech

Perhaps the most interesting result from this pair of studies lies in the open comments of the users, which gives us insight into how successful speech is in terms of acceptance by the general public as a mainstream method of text input. As found in both the lab and ESM studies, participants prefer speech and keyboard for a completely different set of reasons.

Speech takes preference when convenience and constraint become the major deciding factors. Participants mentioned they would choose speech as an input method if have an impairment that would not let them type, or when another task takes away some of the elements that are required for effective typing such as having both hands, visual feedback, and focus.

Participants seemed to be neutral – or at least not significantly biased towards either input method – in terms of entry rate. They found both speech and keyboard to be satisfactorily fast enough for mobile text input.

Typing takes preference when accuracy and social concerns become the major deciding factors. Participants mostly complained about speech not performing well under OOV's, editing what is already been typed, and not identifying specific accents correctly. Further almost all the participants were concerned that using speech in public places would draw unwanted attention, and would violate their privacy – i.e. not wanting to disturb others or not let others know what they are communicating. A few participants also mentioned that if they were in a situation where they had to enter speech using text, they would rather send an audio clip recording – as supported on many messaging apps such as WhatsApp and Facebook Messenger – or simply start a phone call instead. In the ESM study, when the accuracy and social factors become a concern more often than not – as the users would find themselves in more hostile environments with less network connectivity, higher background noise, or in the presence of other people i.e. in an office, library, cinema, metro, or other crowded places, was a clear "repellent" for the participants with regards to speech input.

7.5 Limitations & Future Work

There are a number of limitations that we have in these experiments, we would like to point them out in this part so that researchers who wish to stem from this work can make provision to remove those eliminations and expand their future work.

7.5.1 Controlling Hand Posture

We did not control the hand posture. Rather we let users use their preferred hand posture and self-report. Therefore we cannot make claims about how hand posture results in a better of inferior user experience with a statistical backing. Users could have varied their hand posture throughout the sessions and reported on the posture they used most frequently. Future studies can attempt to control hand postures for a better understanding.

7.5.2 Controlled Degree of Movement

A controlled experiment could be run in a lab environment where the participant is allowed to move in a controlled manner, and which the movement degree is controlled. These can vary from having the participant do a separate task whilst having to answer a text or do a transcription task on the mobile device, or whilst talking to someone – i.e. perhaps answering a set of questions, or attaining different postures such as standing, sitting, walking, lying down, or carrying something in one hand while typing on the other. The walking speed could perhaps be controlled via a treadmill. By doing so in future, researchers would be able to understand the implications of movement on text entry performance.

7.5.3 ESM Meta Data

Another limitation is that we did not capture the user's incidental movement at the point of ESM sampling. This could be possible by reading motion sensor data from the mobile phones i.e. gyro-meter and accelerometer data. This could also be complemented by GPS location data from the mobile device. Further, modern mobile devices are equipped with sensors that can measure a user's health and fitness data, such as heart rate, the amount of calories burned, steps walked, stairs climbed and so on – analogous to a heart rate monitor of Fitbit device.

A future ESM study could better leverage the device sensors to provide a more comprehensive understanding of users' activities before, during, and after they complete an ESM task. This data in conjunction with the typing performance stats could be used for further insight on how users perform during different exhaustion and energy levels.

7.5.4 Control Typing Outside the Sessions

We only asked the users to self-report on how much they have typed outside the sessions and what methods they used to enter text – i.e. STK, SGK, or Speech, but we didn't attempt to control them. A future study could actually control the typing and speaking outside the sessions, by providing incentives to the participants, and having a mechanism of capturing whether the users actually adhered to the required methods.

7.5.5 Controlling ESM Device Type

We let the users use their own devices, thus bringing in a plethora of devices with different form factors into the study. A future study could attempt to standardize this by providing devices to the participants. However, these devices must be the primary device of the user, this means transferring all the accounts, authentication and logins, including their apps and personal data to the provided device by the researchers, which might raise serious problems about data privacy and ethics. However, future studies could attempt this to provide more control and uniformity in the performance and user experience results.

7.5.6 Transcription Task

In our studies we chose to focus on internal validity and our experimental design only investigated people being seated while writing using a transcription task. While this is by far the most common method to evaluate mobile text entry methods, it is certainly possible to consider variations in which participants walk around, or compose their own text rather than copy stimulus phrases. The task we used was a transcription task, but we think that there can be more comprehensive and realistic tasks that could be used instead. The users could be asked a question instead and the answer could be treated as a response phrase. Calculating entry rate here is pretty straightforward but here the research question arises on what is the stimuli phrase and how to calculate error rate.

7.5.7 More Diverse Population & Larger Samples

We could recruit participants from a wider demographic base e.g. just native English speakers, and wider experience distribution e.g. having at least 2 years of experience in using a software keyboard, and at least 1 year experience in gesture typing etc. Participants could be screened to ensure whether they are at a certain level before recruiting them. Another alternative approach is to release the ESM-inspired tools on an app market and recruit tens of thousands of users, possibly by gamifying the experience.

7.5.8 Experiment Duration

We could have also run the experiment for a longer duration (i.e. 3 months), and spaced out the frequency of having the user to do a task, which we believe could have yielded better and more realistic results.

7.5.9 Accents

In the speech related studies, we did recruit participants from a diverse set of background giving to different dialects and accents when speaking English. However, since we did not attempt to recruit specific groups with specific accents or attempt to counter balance the groups; we did not report any correlational statistics or results based on how speech performance varied with accents. A future study could control for different accents and see how these affect the performance of speech.

7.5.10 Network Latency

It is known that while prediction and error correction for typing in Google Keyboard is done in device, speech recognition occurs in the cloud. Therefore maintaining a decent internet connection is vital for proper speech recognition. In our speech based lab study (Chapter 5), we maintained a constant internet connecting using the super-speed WIFI network Eduroam which was provided by the University of St Andrews. However, we identify two limitations in this approach. Firstly, we did not continually monitor the Wi-Fi connectivity (especially in the ESM study – Chapter 6) – and therefore we do not know if certain attempts to transcribe a given sentence using speech were hindered due to the network latency or the speech recognition itself. Secondly, we did not control for the different levels of connectivity and observe how speech recognition would perform in each situation.

Future studies could take this into consideration by either, having participants transcribe messages using speech over varying levels of network connectivity – i.e. G, 3G, H, H+, 4G, etc. And monitor the outcome of speech-to-text, OR, collect the users voice e.g. via a recording and perform offline experiments on decoding these audio files over varying levels of network connectivity. Either of these could shed more insights on how speech would respond to varying levels of network connectivity. A simpler approach for the ESM study would be collect the connectivity related Meta-data at each sampling point right before and after the transcription task using speech takes place.

7.5.11 Background Noise

It is an accepted that background noise is a major factor affecting speech recognition. In our lab study, we provided a fully quiet, zero background noise environment for the participants. Therefore (in combination with the super-speed Wi-Fi connectivity) what we captured was probably the upper bound performance of speech recognition using Google Speech Engine. In our ESM study, we let the users participate in the study from wherever they may be, and they could have found themselves in a variety of situations with varying background noise.

Again, future studies could address these limitations by (a) in the lab study, exposure the users to varying levels of background noise – by playing music or other representative noises in the background e.g. construction work, a busy street, nature sounds, an airport etc. (b) the participants' attempts at transcribing the phrases through speech could be recorded in a quiet background, and then introduced to varying levels of background noise using audio mixing software i.e. PC-DJ. Then these sound clips with varying background noise could be fed into the decoder to investigate how well speech recognition would perform. (c) for the ESM study, if the background noise at the before, after or during (if the phone has dual microphones which could independently accessed via an API) the ESM sample could be measured, this could provide more insights into how background noise affects speech input.

7.5.12 Unsupervised Learning

Another limitation we identified is the continuous learning nature of the Google Speech Engine. Though it is unclear on how this exactly works, we decided not to address this issue/limitation/confound in this set of studies. For the lab study, we simply used the same device to preserve internal validity across all the participants. The phone was not logged into a particular Google account. However, it is not clear whether the usage of speech by one participant actually affected the next one. In future studies this could be addressed in a number of ways – one possible suggestion for the lab study would be to use different devices of the same make and model, logged into completely different and newly created Google accounts and be used for the experiment. For the ESM study, in conjunction with controlling the ESM device type (above), new Google accounts could be created, logged into and prepared for the participants to run the speech component of the study.

7.5.13 Open choice between Speech and Typing

Beyond the experimental treatments provided in these studies, future studies could incorporate an "open" condition, where participants could choose which input modality they want to use for the task at hand, from the ones given e.g. typing, or speech. This would be more suitable for a ESM study. Based on the circumstances the study could capture the choice they make, the performance, and the reasons behind their choice via a quick survey through the ESM app itself.

Further, this study could be complemented with feed-forward recommendations from the app based on the other ESM data that could be captured e.g. background noise, network latency, gyro and accelerometer data. Then the results from both these conditions – with and without feedforward – could be analysed to find out if providing recommendations to the users results in better performance and user experience when typing on mobile devices.

7.5.13.1 Editing and Correcting Text

Since there were so many comments and feedback about editing and correcting text, a future study could shed more light on this by asking participants to correct existing pieces of text – i.e. replace a word with another, fix a spelling mistake etc. using available corrections options provided in the state-of-the-art. Some of these are, use of the suggestions bar, the platform provided spell-checker (on Android), and features such as Gesture Delete and Gesture Cursor Control available on Google Keyboard.

7.5.14 The Receiving End

Although this is not exactly a study that investigates text entry on mobile devices, this is a possible follow up study that stems from one of the open comments obtained via this work. When investigating the adoption of speech in day to day life – a few users claimed, if they were in the position to use their voice instead of typing, they would rather send a voice clip over a messaging app rather than using speech-to-text. It would be interesting to understand how this would affect the message receiver's user experience i.e. what are the factors affecting the user experience of someone receiving a text, a voice message or being "forced" into a two way conversation i.e. a phone call in an ESM setting.

7.6 Design Implications & Recommendations

Whilst the potential application of these findings is largely up to the reader, we wish to provide a few pointers into what they may be used for. Those planning to develop and release the next state of the art text input system for mobile devices, now have benchmark entry and error rate values to work with.

We suggest that future researchers use these results as a starting point to dwell deeper into strengthening our understanding of what makes a positive mobile text input experience. Future researchers could well perform more meticulous, focused experiments both in the lab and the wild and control for factors such as percentage of OOV words, sentence perplexity, hand posture, device form factor, degree of movement, background noise and network latency.

Researchers should definitely step out of the lab to find out what really impacts a user's text entry experience. Nearly all text entry studies are based on controlled lab experiments like our lab studies, and we believe that while such studies do provide valuable insight, it is only half the story. Researchers need to conduct both lab and ESM studies when evaluating text input methods as users find themselves in both situations.

7.6.1 Providing Recommendations for the User

Users should be educated further on how to obtain the best of all worlds (STK, SGK and speech input). They should be educated on how which hand postures are best for which situation, and if using SGK or speech input, on how to quickly identify a word not being recognized is due to OOV and switch to STK instead of performing repeated gestures or multiple attempts at speaking the same word, all these will improve a user's text input experience.

What happens at the moment is that an average user would install a novel text input system on their mobile device, and will be subject to trial and error it, which might not even reveal certain features which are actually provided to make their life easier. Also they might not be using the most optimal input method for a given scenario they might find themselves in – i.e. not switching to speech when driving (which is dangerous and illegal), not switching to SGK when performing one handed typing because your other hand is busy, not immediately switching to STK when a word is misspelt or not successfully entered in the first go.

IME's that support both STK, SGK and speech methods of input can now provide suggestions to the users based on the location and scenario they are in. It was seen that the users do not make full use of the proper method of input or proper posture for their situation unless they are made aware of it – now the IME software itself can notify the user based on detecting movement or incline, to switch their hand posture or input method for a seamless experience.

7.6.2 Recommendations for Handling OOVs better

IME developers can work towards learning contextual phrases and words from the application in focus, to minimize the errors caused by OOV words when using SGK and speech input.

SGK was definitely more appealing to users, but it can be seen that in controlled environments STK outperforms SGK. In the same way, speech was also more appealing to users for its convenience, yet suffered from the similar problems as SGK with regards to error correction and entering OOVs. SGK or speech input cannot stand on their own due to not being able to handle the OOVs and needs to be complemented with STK, as it is now. At the moment pure SGK users or pure speech input users do not have a way of reverting back to the original typed word and ignore the guessed word, as STK users do.

Our recommendation is that both speech input and SGK should be improved to handle OOV's better, as this is now the main bottleneck for the state of the art. Personalization could be one such solution, which would provide better guesses in the user's domain. SGK should have a joint multigram model that infers non-dictionary words when users are tracing from key to key, or to have one small dictionary that proposes common words and a large dictionary (say > 1 million words) that identifies OOV words, and let the user select the OOV word from the n-best list. Once selected, the OOV word can be set as active.

7.6.3 Recommendations for Speech

Hardware (i.e. phone manufacturers) and OS (e.g. Google for Android) providers and software vendors should work together in enhancing the overall speech experience for the user e.g. to decode speech better in low network environments by performing more decoding on-device, filtering background noise better using noise cancellation techniques – so that the user can speak more discreetly in public spaces without drawing attention to oneself and violating their privacy. This would help users adopt speech more as a day to day text entry mechanism.

8
Conclusions

As per our motivation as set out in the Introduction chapter, we set out to explore this central research idea in this thesis:

> **"Controlled experimental A/B lab comparisons and ESM-based in-the-wild studies both inform similar and complementary aspects of the text entry experience. However, when used in conjunction, they are capable of more comprehensively assessing the complete text entry user experience."**

By answering the following research questions:

1. What are the factors that affect the performance and user experience of mobile text entry using a Smart Touch Keyboard (STK) and a Smart Gesture Keyboard (SGK) in a lab setting?
   In chapter 3, we have answered this question by running a lab based longitudinal study.

2. What are the factors that affect the performance and user experience of mobile text entry using a Smart Touch Keyboard (STK) and a Smart Gesture Keyboard (SGK) outside a lab setting a.k.a. in the wild
   In chapter 4, we have answered this question by running a novel ESM based study.

3. What are the factors that affect the performance and user experience of mobile text entry using a keyboard vs speech input in a lab setting?
   In chapter 5, we have answered this question by running longitudinal lab study.

4. What are the factors that affect the performance and user experience of mobile text entry using a keyboard vs speech input outside a lab setting a.k.a. in the wild
   In chapter 6, we have answered this question by running an ESM based study; and will continue to run until the expected number of data points is collected.

As described in detail in the Literature Review and Discussion chapters, we understand that studying mobile text entry is challenging. As more innovation happens in the text entry field, evolving from physical keyboards to software keyboards, that continually evolve to perform better prediction, error correction, and provide mechanisms for increased speed and fewer errors such as optimised layouts, language and special model based decoding, and gesture keyboards, we find that most reported text entry research has been based on research prototypes. Continued progress and innovation in the text entry field cannot have a solid empirical footing if we do not know how well current technologies work for user a.k.a. the current state of the art.

This thesis aims to explore this gap in detail and shed light on this aspect by the proposition of a research methodology that is a better model of reality. An experiment should be a model of reality, and the existing problem is that the very common “controlled experiment / transcription task” model doesn’t paint a very realistic picture, and therefore isn’t a realistic model. The proposed solution is that while understanding that any model is not a perfect model, and thus cannot paint an exact picture of reality, this thesis explores modifications of the existing methodologies to help piece together a better picture – thus we do not aim to propose a perfect model, but a better one.

Further this thesis makes a number of contributions to the field of text entry, namely:

- The perspective of complimenting traditional text entry studies with in-situ empirical data, to understand text entry better

- The first study (and CHI publication) that empirically investigates the performance and user experience of gesture keyboard in any setting
- The first study (and CHI publication) that presents an ESM based "in-the-wild" comparison of typing vs gesture on the state-of-the-art
- The first speech study that investigates empirical data in and outside the lab with varying levels of sentence complexity
- The first in-situ speech study outside the lab with the state-of-the-art
- Software for carrying out lab and ESM studies – it must be noted here that other products are available at the time of submission of this thesis (2018), but at the time of running the studies A&B (2013) there were no software for this specific requirement and therefore it was required that own experimental software be developed from scratch

Despite the prevalence of STKs, SGKs, and Speech Based Input, it was evident in the literature review that there is a clear lack of in-depth studies about their text entry performance, in particular outside a lab environment. Therefore empirical research to-date has been limited in scope, size, and technology form factor.

From the results presented in this thesis, in Chapters 3-4, we are now able to make conclusions on the performance and user experience of STK's and SGK's, such as:

- STK's are faster in controlled environments due to the controlled environment favouring STK's more, however with practice users tend to become faster with SGK's over time

- SGK's produce more errors due to the presence of OOV words and users have no way of correcting these errors without switching the method of input

- Hand postures are a definite confounding variable that needs to be controlled and explored further, though the results are indicative towards two-thumb STK's perform best in a lab setting, and single-thumb-SGK's perform best in the wild

- When exposed to it, users tend to migrate from STK to SGK, which was very interesting to observe

Further, from the results obtained in Chapters 5-6, we are now able to make conclusions on the performance and user experience of STK's and SGK's such as:

- Speech performs better in controlled environments as there is no background noise, no in-device distractions, having full network connectivity and users have full focus and visual feedback on what they are trying to enter

- Speech produces more errors, suffering from the same shortcomings that SGK's suffered from, without having support for OOV's, with the added performance hit due to the hostile conditions mentioned overleaf. We now understand that to have a positive speech based text entry experience, these factors need to be accounted for when designing novel speech based text entry mechanisms

- We understand that sentence complexity and OOVs effects speech more in a controlled lab based environment, but affects typing more outside a lab environment – in which speech maybe a better choice when considering performance alone

- Finally we now understand better that users adopting speech as a default mechanism of text entry is a complex decision, with factors such as speed, convenience and constraints being in favour of speech being chosen, and accuracy and social concerns being against speech being chosen

From both sets of conclusions above, it is evident that albeit having faster performance in many controlled and uncontrolled situations, to have a better user experience in terms of gesture keyboard and speech, researchers need to handle OOV's better, and focus on personalisation and better context based prediction, especially with regards to how speech performance varies when we vary the sentence complexity.

Furthermore, I am a firm believer of replication. It is apparent with advances in technology, the current hardware and software will be replaced with better performing, more advanced versions in a matter of years. Novel paradigms such as Machine Learning are changing the way we approach problems and come up with solutions. Therefore, the results presented in this thesis may have to be revisited in a matter of years, with the looming advancement of hardware, software and technology. For more nuanced views of the text entry experience I have concluded the Discussion with a number of avenues that researchers could take in running future text entry studies and making developments.

Finally, the research methodology matters. It is very clear that we could obtain both complimentary and contradictory evidence when we carry out studies in the wild vs studies in the lab – thus enhancing our understanding. A good example for this is the STK/SGK studies – A& B – a naïve lab study alone would not reflect these findings and thus it was not known before the paper was published – thus bringing us to our final conclusion:

Studies in the lab and in the wild both can be informative to different aspects of keyboard experience, but used in conjunction is more reliable in comprehensively assessing input technologies of current and future generations.

9

References

AndroidCentral. (2012). LG Nexus 4 Specifications. Retrieved October 1, 2018, from https://www.androidcentral.com/lg-nexus-4-specs

Bassil, Y., & Alwani, M. (2012). Post-Editing Error Correction Algorithm for Speech Recognition using Bing Spelling Suggestion. Retrieved from https://arxiv.org/abs/1203.5255

Baudisch, P., & Chu, G. (2006). *Back-of-Device Interaction Allows Creating Very Small Touch Devices*. Retrieved from http://www.itu.dk/~tped/teaching/pervasive/SPCT-F2015/baudisch_and_chu_2009.pdf

Bi, X., Chelba, C., Ouyang, T., Partridge, K., & Zhai, S. (2012). Bimanual gesture keyboard. In *Proceedings of the 25th annual ACM symposium on User interface software and technology - UIST '12* (p. 137). New York, New York, USA: ACM Press. https://doi.org/10.1145/2380116.2380136

Bi, X., Ouyang, T., Zhai, S., Bi, X., Ouyang, T., & Zhai, S. (2014). Both complete and correct? In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14* (pp. 2297–2306). New York, New York, USA: ACM Press. https://doi.org/10.1145/2556288.2557414

Bi, X., Smith, B. A., & Zhai, S. (2010). Quasi-qwerty soft keyboard optimization. In *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10* (p. 283). New York, New York, USA: ACM Press.

https://doi.org/10.1145/1753326.1753367

Bi, X., & Zhai, S. (2016). IJQwerty: What Difference Does One Key Change Make? Gesture Typing Keyboard Optimization Bounded by One Key Position Change from Qwerty. https://doi.org/10.1145/2858036.2858421

Brewster, S., Chohan, F., Brown, L., & Brown, L. (2007). Tactile feedback for mobile interactions. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07* (p. 159). New York, New York, USA: ACM Press. https://doi.org/10.1145/1240624.1240649

Castellucci, S. J., & MacKenzie, I. S. (2011). Gathering text entry metrics on android devices. In *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems - CHI EA '11* (p. 1507). New York, New York, USA: ACM Press. https://doi.org/10.1145/1979742.1979799

Cerney, M. M., Mila, B. D., & Hill, L. C. (2004). Comparison of Mobile Text Entry Methods. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *48*(5), 778–782. https://doi.org/10.1177/154193120404800508

Chen, T., & Kan, M.-Y. (2012). Creating a live, public short message service corpus: the NUS SMS corpus. *Language Resources and Evaluation*, *47*(2), 299–335. https://doi.org/10.1007/s10579-012-9197-9

Clawson, J., Lyons, K., Starner, T., & Clarkson, E. (2005). The Impacts of Limited Visual Feedback on Mobile Text Entry for the Twiddler and Mini-QWERTY Keyboards. In *Ninth IEEE International Symposium on Wearable Computers (ISWC'05)* (pp. 170–177). IEEE. https://doi.org/10.1109/ISWC.2005.49

Consolvo, S., & Walker, M. (2003). Using the experience sampling method to evaluate ubicomp applications. *IEEE Pervasive Computing*, *2*(2), 24–31. https://doi.org/10.1109/MPRV.2003.1203750

Curran, K., Woods, D., & Riordan, B. O. (2006). Investigating text input methods for mobile phones. *Telematics and Informatics*, *23*(1), 1–21. https://doi.org/10.1016/J.TELE.2004.12.001

Dhakal, V., Feit, A. M., Kristensson, P. O., & Oulasvirta, A. (2018). Observations on Typing from 136 Million Keystrokes. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18* (pp. 1–12). New York, New York, USA: ACM Press. https://doi.org/10.1145/3173574.3174220

Dunlop, M. D., & Crossan, A. (1999). Dictionary based text entry method for mobile phones. Retrieved from https://strathprints.strath.ac.uk/32316/

Dunlop, M., & Levine, J. (2012). Multidimensional pareto optimization of touchscreen keyboards for speed, familiarity and improved spell checking. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12* (p. 2669). New York, New York, USA: ACM Press. https://doi.org/10.1145/2207676.2208659

E.161 : Arrangement of digits, letters and symbols on telephones and other devices that can be used for gaining access to a telephone network. (2001). Retrieved January 26, 2019, from https://www.itu.int/rec/T-REC-E.161-200102-I/en

Endgadget. (n.d.). T9 Trace. Retrieved October 1, 2018, from https://www.engadget.com/2010/03/24/t9-trace-lets-you-swype-through-your-text-messages/

Fisher, R. A., & Yates, F. (1948). *Statistical tables for biological, agricultural and medical research* (3rd ed.). London: Oliver & Boyd.

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, *47*(6), 381–391. https://doi.org/10.1037/h0055392

Gizmodo. (2014). Why we still use QWERTY. Retrieved October 1, 2018, from https://gizmodo.com/why-we-still-use-qwerty-keyboards-even-though-theyre-a-1643855077

Goel, M., Findlater, L., & Wobbrock, J. (2012). WalkType. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12* (p. 2687). New York, New York, USA: ACM Press.

https://doi.org/10.1145/2207676.2208662

Goel, M., Jansen, A., Mandel, T., Patel, S. N., & Wobbrock, J. O. (2013). ContextType. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13* (p. 2795). New York, New York, USA: ACM Press. https://doi.org/10.1145/2470654.2481386

Goodman, J., Venolia, G., Steury, K., & Parker, C. (2002). Language modeling for soft keyboards. In *Proceedings of the 7th international conference on Intelligent user interfaces - IUI '02* (p. 194). New York, New York, USA: ACM Press. https://doi.org/10.1145/502716.502753

Google. (2018a). Android Auto. Retrieved October 1, 2018, from https://www.android.com/auto/

Google. (2018b). Android Open Source Project. Retrieved October 1, 2018, from https://source.android.com/

Google. (2018c). Google Keyboard. Retrieved October 1, 2018, from https://play.google.com/store/apps/details?id=com.google.android.inputmethod.latin&hl=en_GB

Google. (2018d). SpeechRecognizer | Android Developers. Retrieved October 2, 2018, from https://developer.android.com/reference/android/speech/SpeechRecognizer

GSMArena. (2018). Huawei P20 lite - Full phone specifications. Retrieved October 2, 2018, from https://www.gsmarena.com/huawei_p20_lite-9098.php

H Fischer, A. R., Price, K. J., Sears, A., & Price Andrew Sears, K. J. (2005). Speech-Based Text Entry for Mobile Handheld Devices: An Analysis of Efficacy and Error Correction Techniques for Server-Based Solutions. *International Journal of Human-Computer Interaction, 19*(3), 279–304. https://doi.org/10.1207/s15327590ijhc1903_1

Helander, M., Landauer, T. K., & Prabhu, P. V. (1997). *Handbook of human-computer interaction*. Elsevier.

Henze, N., Rukzio, E., & Boll, S. (2012). Observational and experimental investigation of typing behaviour using virtual keyboards for mobile devices. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI ’12* (p. 2659). New York, New York, USA: ACM Press. https://doi.org/10.1145/2207676.2208658

Hiraga, Y., & Ono, Y. (1980). *AN ANALYS1S OF THE STANDARD ENGLISH KEYBOARD*. Retrieved from http://www.aclweb.org/anthology/C80-1036

IELTS. (n.d.). Retrieved January 26, 2019, from https://www.ielts.org/

ISO 9241-210:2010 - Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems. (2010). Retrieved January 25, 2019, from https://www.iso.org/standard/52075.html

Isokoski, P., & Raisamo, R. (2000). Device independent text input. In *Proceedings of the working conference on Advanced visual interfaces - AVI ’00* (pp. 76–83). New York, New York, USA: ACM Press. https://doi.org/10.1145/345513.345262

James, C. L., & Reischel, K. M. (2001). Text input for mobile devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI ’01* (pp. 365–371). New York, New York, USA: ACM Press. https://doi.org/10.1145/365024.365300

Jurafsky, D., & Martin, J. H. (2009). *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition*. Pearson Prentice Hall.

Kano, A., & He, P. (n.d.). Evaluating Phrase Sets for Use with Text Entry Method Evaluation with Dyslexic Participants. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.97.4741

Kano, A., Read, J. C., & Dix, A. (2006). Children’s phrase set for text input method evaluations. In *Proceedings of the 4th Nordic conference on Human-computer interaction changing roles - NordiCHI ’06* (pp. 449–452). New York, New York, USA: ACM Press. https://doi.org/10.1145/1182475.1182534

Karat, C.-M., Halverson, C., Horn, D., & Karat, J. (1999). Patterns of entry and correction in large vocabulary continuous speech recognition systems. In *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99* (pp. 568–575). New York, New York, USA: ACM Press. https://doi.org/10.1145/302979.303160

Keith Vertnanen. (n.d.). The Enron Mobile Email Dataset. Retrieved September 30, 2018, from https://www.keithv.com/software/enronmobile/

Kjeldskov, J., & Skov, M. B. (2014). Was it worth the hassle? In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services - MobileHCI '14* (pp. 43–52). New York, New York, USA: ACM Press. https://doi.org/10.1145/2628363.2628398

Kristensson, P.-O. (2007). *Discrete and continuous shape wrtiting for text entry and control*. Linköping University. Retrieved from http://swepub.kb.se/bib/swepub:oai:DiVA.org:liu-8877?tab2=abs&language=en

Kristensson, P.-O., & Vertanen, K. (2010). Asynchronous Multimodal Text Entry using Speech and Gesture Keyboards. *Proceedings of the International Conference on Spoken Language Processing, August*, 1698--1701.

Kristensson, P.-O., & Zhai, S. (2004). SHARK ^{2}. In *Proceedings of the 17th annual ACM symposium on User interface software and technology - UIST '04* (p. 43). New York, New York, USA: ACM Press. https://doi.org/10.1145/1029632.1029640

Kristensson, P.-O., & Zhai, S. (2005). Relaxing stylus typing precision by geometric pattern matching. In *Proceedings of the 10th international conference on Intelligent user interfaces - IUI '05* (p. 151). New York, New York, USA: ACM Press. https://doi.org/10.1145/1040830.1040867

Kristensson, P. O. (2009). Five Challenges for Intelligent Text Entry Methods. *AI Magazine*, *30*(4), 85. https://doi.org/10.1609/aimag.v30i4.2269

Kristensson, P. O. (2011). Design dimensions of intelligent text entry tutors. *AIED'11*

Proceedings of the 15th International Conference on Artificial Intelligence in Education, 494–496. Retrieved from https://dl.acm.org/citation.cfm?id=2026589

Kristensson, P. O., & Denby, L. (2009). Text entry performance of state of the art unconstrained handwriting recognition: a longitudinal user study. *Techniques*, 567–570. Retrieved from http://eprints.pascal-network.org/archive/00005605/

Kristensson, P. O., & Vertanen, K. (2012). *Performance Comparisons of Phrase Sets and Presentation Styles for Text Entry Evaluations*. Retrieved from http://wing.comp.nus.edu.sg/SMSCorpus/

Kristensson, P. O., & Vertanen, K. (2014). The inviscid text entry rate and its application as a grand goal for mobile text entry. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services - MobileHCI '14* (pp. 335–338). New York, New York, USA: ACM Press. https://doi.org/10.1145/2628363.2628405

Kristensson, P. O., & Zhai, S. (2007). Command strokes with and without preview. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07* (p. 1137). New York, New York, USA: ACM Press. https://doi.org/10.1145/1240624.1240797

Kristensson, P. O., & Zhai, S. (2008). Improving word-recognizers using an interactive lexicon with active and passive words. In *Proceedings of the 13th international conference on Intelligent user interfaces - IUI '08* (p. 353). New York, New York, USA: ACM Press. https://doi.org/10.1145/1378773.1378828

Levenshtein, V. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals | BibSonomy. *Soviet Physics Doklady*, *10*, 707. Retrieved from https://www.bibsonomy.org/bibtex/21a29b294552edb63828d57f3495e2eb2/brightbyte

Levine, S. H., & Goodenough-Trepagnier, C. (1990). Customised text entry devices for motor-impaired users. *Applied Ergonomics*, *21*(1), 55–62. https://doi.org/10.1016/0003-6870(90)90074-8

Lewis, J. R., LaLomia, M. J., & Kennedy, P. J. (1999). Evaluation of Typing Key Layouts for Stylus Input. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *43*(5), 420–424. https://doi.org/10.1177/154193129904300506

MacKenzie, I. S. (2002). KSPC (Keystrokes per Character) as a Characteristic of Text Entry Techniques (pp. 195–210). Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-45756-9_16

Mackenzie, I. S., & Soukoreff, R. W. (2002). *Text Entry for Mobile Computing: Models and Methods, Theory and Practice* (Vol. 17). Retrieved from http://www.gsmworld.com

MacKenzie, I. S., & Soukoreff, R. W. (2003). Phrase sets for evaluating text entry techniques. In *CHI '03 extended abstracts on Human factors in computing systems - CHI '03* (p. 754). New York, New York, USA: ACM Press. https://doi.org/10.1145/765891.765971

MacKenzie, I. S., & Zhang, S. X. (1999). The design and evaluation of a high-performance soft keyboard. In *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99* (pp. 25–31). New York, New York, USA: ACM Press. https://doi.org/10.1145/302979.302983

Mackenzie, I. S., Zhang, S. X., & Soukoreff, R. W. (1999). Text entry using soft keyboards. *Behaviour & Information Technology*, *18*(4), 235–244. https://doi.org/10.1080/014492999118995

Magnien, L., Bouraoui, J. L., & Vigouroux, N. (2004). Mobile Text Input with Soft Keyboards: Optimization by Means of Visual Clues (pp. 337–341). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-28637-0_33

Mayzner, M. S., & Tresselt, M. (1965). *Tables of Single-letter and Digram Frequency Counts for Various Word-length and Letter-position Combinations*. Psychonomic Press. Retrieved from https://books.google.co.uk/books/about/Tables_of_Single_letter_and_Digram_Fre qu.html?id=FI7BHgAACAAJ&redir_esc=y

Moore, R. K. (2004). Modelling Data Entry Rates for ASR and Alternative Input Methods. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.144.5434

Nel, E.-M., Kristensson, P.-O., & MacKay, D. J. C. (2018). Ticker: An Adaptive Single-Switch Text Entry Method for Visually Impaired Users. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1. https://doi.org/10.1109/TPAMI.2018.2865897

Nguyen, H., & Bartha, M. C. (2012). Shape Writing on Tablets: Better Performance or Better Experience? *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 56*(1), 1591–1593. https://doi.org/10.1177/1071181312561317

Nielson. (n.d.). Mobile Usage Among Teenagers. Retrieved from nielsen.com/us/en/newswire/2010/u-s-teenmobile-report-calling-yesterday-texting-today-usingapps-tomorrow.html

Oulasvirta, A., Reichel, A., Li, W., Zhang, Y., Bachynskyi, M., Vertanen, K., & Kristensson, P. O. (2013). Improving two-thumb text entry on touchscreen devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13* (p. 2765). New York, New York, USA: ACM Press. https://doi.org/10.1145/2470654.2481383

Partridge, K., Chatterjee, S., Sazawal, V., Borriello, G., & Want, R. (2002). *TiltType: Accelerometer-Supported Text Entry for Very Small Devices*. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.103.6600&rep=rep1&type=pdf

Perlin, K., & Ken. (1998). Quikwriting. In *Proceedings of the 11th annual ACM symposium on User interface software and technology - UIST '98* (pp. 215–216). New York, New York, USA: ACM Press. https://doi.org/10.1145/288392.288613

Petajan, E., Bischoff, B., Bodoff, D., & Brooke, N. M. (1988). An improved automatic lipreading system to enhance speech recognition. In *Proceedings of the SIGCHI* *conference on Human factors in computing systems - CHI '88* (pp. 19–25). New York, New York, USA: ACM Press. https://doi.org/10.1145/57167.57170

Reyal, S., Zhai, S., & Kristensson, P. O. (2015). Performance and user experience of touchscreen and gesture keyboards in a lab setting and in the wild. In *Conference on Human Factors in Computing Systems - Proceedings* (Vol. 2015–April). https://doi.org/10.1145/2702123.2702597

Rick, J., & Jochen. (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* (p. 77). New York, New York, USA: ACM Press. https://doi.org/10.1145/1866029.1866043

Rosenbaum, D. A. (1991). *Human motor control*. Academic Press. Retrieved from https://books.google.co.uk/books/about/Human_Motor_Control.html?id=w3kVQBqSlEQC&redir_esc=y

Roudaut, A., Huot, S., & Lecolinet, E. (2008). TapTap and MagStick. In *Proceedings of the working conference on Advanced visual interfaces - AVI '08* (p. 146). New York, New York, USA: ACM Press. https://doi.org/10.1145/1385569.1385594

Rough, D., Vertanen, K., & Kristensson, P. O. (2014). An evaluation of Dasher with a high-performance language model as a gaze communication method. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces - AVI '14* (pp. 169–176). New York, New York, USA: ACM Press. https://doi.org/10.1145/2598153.2598157

Ruan, S., Wobbrock, J. O., Liou, K., Ng, A., & Landay, J. (2016). *Speech Is 3x Faster than Typing for English and Mandarin Text Entry on Mobile Devices*. Retrieved from https://hci.stanford.edu/research/speech/paper/speech_paper.pdf

Shabir, M., Tieng Wei, K., Abd. Ghani, A. A., & Kamaruddin, A. (2015). A literature review on mobile devices touch screen inputs and its techniques evaluation. In *2015 9th Malaysian Software Engineering Conference (MySEC)* (pp. 154–160).

IEEE. https://doi.org/10.1109/MySEC.2015.7475213

Shneiderman, B. (2000). The limits of speech recognition. *Communications of the ACM*, *43*(9), 63–65. https://doi.org/10.1145/348941.348990

Shusterman, R. (2011). Muscle Memory and the Somaesthetic Pathologies of Everyday Life. *Human Movement*, *12*(1).

Sirisena, A. (2002). Mobile text entry. Retrieved from https://ir.canterbury.ac.nz/handle/10092/9554

Smith, B. A., & Zhai, S. (2002). Optimised Virtual Keyboards with and without Alphabetical Ordering - A Novice User Study. *IN PROCEEDINGS OF INTERACT'2001 - IFIP INTERNATIONAL CONFERENCE ON HUMAN-COMPUTER INTERACTION. 2001*, 92--99. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.28.8207

Software multi-tap input system and method. (2003). Retrieved from https://patents.google.com/patent/US20040201576A1/en

Soukoreff, R. W., & MacKenzie, I. S. (2003). Metrics for text entry research. In *Proceedings of the conference on Human factors in computing systems - CHI '03* (p. 113). New York, New York, USA: ACM Press. https://doi.org/10.1145/642611.642632

SwiftKey. (n.d.). Retrieved October 1, 2018, from https://www.microsoft.com/en-us/swiftkey?rtc=1&activetab=pivot_1%3Aprimaryr2

Vertanen, K., Emge, J., Memmi, H., & Kristensson, P. O. (2014). Text blaster. In *Proceedings of the extended abstracts of the 32nd annual ACM conference on Human factors in computing systems - CHI EA '14* (pp. 379–382). New York, New York, USA: ACM Press. https://doi.org/10.1145/2559206.2574802

Vertanen, K., Fletcher, C., Gaines, D., Gould, J., & Kristensson, P. O. (2018). The Impact of Word, Multiple Word, and Sentence Input on Virtual Keyboard Decoding Performance. In *Proceedings of the 2018 CHI Conference on Human*

Factors in Computing Systems - CHI '18 (pp. 1–12). New York, New York, USA: ACM Press. https://doi.org/10.1145/3173574.3174200

Vertanen, K., & Kristensson, P.-O. (2010a). Intelligently Aiding Human-Guided Correction of Speech Recognition. *AAAI '10: Proceedings of the 24th AAAI Conference on Artificial Intelligence*, 1698--1701.

Vertanen, K., & Kristensson, P. O. (2010b). Getting it right the second time: Recognition of spoken corrections. In *2010 IEEE Spoken Language Technology Workshop* (pp. 289–294). IEEE. https://doi.org/10.1109/SLT.2010.5700866

Vertanen, K., & Kristensson, P. O. (2011). A versatile dataset for text entry evaluations based on genuine mobile emails. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services - MobileHCI '11* (p. 295). New York, New York, USA: ACM Press. https://doi.org/10.1145/2037373.2037418

Vertanen, K., & Kristensson, P. O. (2014). Complementing text entry evaluations with a composition task. *ACM Transactions on Computer-Human Interaction*, *21*(2), 1–33. https://doi.org/10.1145/2555691

Vertanen, K., & MacKay, D. J. C. (2014). Speech dasher. In *Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility - ASSETS '14* (pp. 353–354). New York, New York, USA: ACM Press. https://doi.org/10.1145/2661334.2661420

Vertanen, K., Memmi, H., Emge, J., Reyal, S., & Kristensson, P. O. (2015). Velocitap: Investigating fast mobile text entry using sentence-based decoding of touchscreen keyboard input. In *Conference on Human Factors in Computing Systems - Proceedings* (Vol. 2015–April). https://doi.org/10.1145/2702123.2702135

Vertanen, K., Memmi, H., & Kristensson, P. O. (2013). The feasibility of eyes-free touchscreen keyboard typing. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '13* (pp. 1–2). New York, New York, USA: ACM Press.

https://doi.org/10.1145/2513383.2513399

Vertanen, K., Vertanen, K., & Kristensson, P. O. (n.d.). Recognition and correction of voice web search queries. *IN PROC. INTERSPEECH '09*, 1863--1866. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.208.7415

Vogel, D., & Baudisch, P. (2007). *Shift: A Technique for Operating Pen-Based Interfaces Using Touch*. Retrieved from http://www.patrickbaudisch.com/publications/2007-Vogel-CHI07-Shift.pdf

Ward, D. J., Blackwell, A. F., & MacKay, D. J. C. (2000). Dasher---a data entry interface using continuous gestures and language models. In *Proceedings of the 13th annual ACM symposium on User interface software and technology - UIST '00* (pp. 129–137). New York, New York, USA: ACM Press. https://doi.org/10.1145/354401.354427

Weir, D., Pohl, H., Rogers, S., Vertanen, K., Kristensson, P. O., Weir, D., … Kristensson, P. O. (2014). Uncertain text entry on mobile devices. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14* (pp. 2307–2316). New York, New York, USA: ACM Press. https://doi.org/10.1145/2556288.2557412

Weir, D., Rogers, S., Murray-Smith, R., & Löchtefeld, M. (2012). A user-specific machine learning approach for improving touch accuracy on mobile devices. In *Proceedings of the 25th annual ACM symposium on User interface software and technology - UIST '12* (p. 465). New York, New York, USA: ACM Press. https://doi.org/10.1145/2380116.2380175

Wigdor, D., & Balakrishnan, R. (2002). *Empirical Investigation into the Effect of Orientation on Text Readability in Tabletop Displays*. Retrieved from http://www.dgp.toronto.edu/~dwigdor/research/wigdor_ecscw_2005.pdf

Wilson, G., Brewster, S. A., Halvey, M., Crossan, A., & Stewart, C. (2011). The effects of walking, feedback and control method on pressure-based interaction. In *Proceedings of the 13th International Conference on Human Computer Interaction* *with Mobile Devices and Services - MobileHCI ’11* (p. 147). New York, New York, USA: ACM Press. https://doi.org/10.1145/2037373.2037397

Yeo, H.-S., Phang, X.-S., Castellucci, S. J., Kristensson, P. O., & Quigley, A. (2017). Investigating Tilt-based Gesture Keyboard Entry for Single-Handed Text Entry on Large Devices. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI ’17* (pp. 4194–4202). New York, New York, USA: ACM Press. https://doi.org/10.1145/3025453.3025520

Zhai, S., Hunter, M., & Smith, B. (2002). Performance optimization of virtual keyboards. *Taylor & Francis*. Retrieved from https://www.tandfonline.com/doi/abs/10.1080/07370024.2002.9667315

Zhai, S., & Kristensson, P.-O. (2003). Shorthand writing on stylus keyboard. In *Proceedings of the conference on Human factors in computing systems - CHI ’03* (p. 97). New York, New York, USA: ACM Press. https://doi.org/10.1145/642611.642630

Zhai, S., & Kristensson, P. O. (2008). Interlaced QWERTY. In *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI ’08* (p. 593). New York, New York, USA: ACM Press. https://doi.org/10.1145/1357054.1357149

Zhai, S., & Kristensson, P. O. (2012). The word-gesture keyboard. *Communications of the ACM*, *55*(9), 91. https://doi.org/10.1145/2330667.2330689

Zhai, S., Sue, A., & Accot, J. (2002). Movement model, hits distribution and learning in virtual keyboarding. In *Proceedings of the SIGCHI conference on Human factors in computing systems Changing our world, changing ourselves - CHI ’02* (p. 17). New York, New York, USA: ACM Press. https://doi.org/10.1145/503376.503381