

Syracuse University

**SURFACE**

---

Dissertations - ALL

SURFACE

---

December 2020

# USER AUTHENTICATION ACROSS DEVICES, MODALITIES AND REPRESENTATION: BEHAVIORAL BIOMETRIC METHODS

Amith Kamath Kamath Belman  
*Syracuse University*

Follow this and additional works at: <https://surface.syr.edu/etd>



Part of the [Engineering Commons](#)

---

## Recommended Citation

Kamath Belman, Amith Kamath, "USER AUTHENTICATION ACROSS DEVICES, MODALITIES AND REPRESENTATION: BEHAVIORAL BIOMETRIC METHODS" (2020). *Dissertations - ALL*. 1228.  
<https://surface.syr.edu/etd/1228>

This Dissertation is brought to you for free and open access by the SURFACE at SURFACE. It has been accepted for inclusion in Dissertations - ALL by an authorized administrator of SURFACE. For more information, please contact [surface@syr.edu](mailto:surface@syr.edu).

## ABSTRACT

Biometrics eliminate the need for a person to remember and reproduce complex secretive information or carry additional hardware in order to authenticate oneself. Behavioral biometrics is a branch of biometrics that focuses on using a person's behavior or way of doing a task as means of authentication. These tasks can be any common, day to day tasks like walking, sleeping, talking, typing and so on. As interactions with computers and other smart-devices like phones and tablets have become an essential part of modern life, a person's style of interaction with them can be used as a powerful means of behavioral biometrics.

In this dissertation, we present insights from the analysis of our proposed set of context-sensitive or word-specific keystroke features on desktop, tablet and phone. We show that the conventional features are not highly discriminatory on desktops and are only marginally better on hand-held devices for user identification. By using information of the context, our proposed word-specific features offer superior discrimination among users on all devices. Classifiers, built using our proposed features, perform user identification with high accuracies in range of 90% to 97%, average precision and recall values of 0.914 and 0.901 respectively. Analysis of the word-based impact factors reveal that four or five character words, words with about 50% vowels, and those that are ranked higher on the frequency lists might give better results for the extraction and use of the proposed features for user identification.

We also examine a large umbrella of behavioral biometric data such as; keystroke latencies, gait and swipe data on desktop, phone and tablet for the assumption of an underlying normal distribution, which is common in many research works. Using suitable non-parametric normality tests (Lilliefors test and Shapiro-Wilk test) we show that a majority of the features from all activities and all devices, do not follow a normal distribution. In

most cases less than 25% of the samples that were tested had p values  $> 0.05$ . We discuss alternate solutions to address the non-normality in behavioral biometric data.

Openly available datasets did not provide the wide range of modalities and activities required for our research. Therefore, we have collected and shared an open access, large benchmark dataset for behavioral biometrics on IEEEDataport. We describe the collection and analysis of our *Syracuse University and Assured Information Security - Behavioral Biometrics Multi-device and multi -Activity data from Same users (SU-AIS BB-MAS) Dataset*. Which is an open access dataset on IEEEdataport, with data from 117 subjects for typing (both fixed and free text), gait (walking, upstairs and downstairs) and touch on Desktop, Tablet and Phone. The dataset consists a total of about: 3.5 million keystroke events; 57.1 million data-points for accelerometer and gyroscope each; 1.7 million data-points for swipes and is listed as one of the most popular datasets on the portal (through IEEE emails to all members on 05/13/2020 and 07/21/2020).

We also show that keystroke dynamics (KD) on a desktop can be used to classify the type of activity, either benign or adversarial, that a text sample originates from. We show the inefficiencies of popular temporal features for this task. With our proposed set of 14 features we achieve high accuracies (93% to 97%) and low Type 1 and Type 2 errors (3% to 8%) in classifying text samples of different sizes. We also present exploratory research in (a) authenticating users through musical notes generated by mapping their keystroke latencies to music and (b) authenticating users through the relationship between their keystroke latencies on multiple devices.

USER AUTHENTICATION ACROSS DEVICES, MODALITIES AND  
REPRESENTATION: BEHAVIORAL BIOMETRIC METHODS

by

Amith Kamath Belman

B.E., Visveswaraya Technological University, 2011

M.Tech., Visveswaraya Technological University, 2013

Submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Computer & Information Science and Engineering.

Syracuse University

December 2020

Copyright © Amith Kamath Belman 2020

All Rights Reserved

To my family.

## ACKNOWLEDGMENTS

I am grateful to my advisor, Prof. Vir V. Phoha for giving me the opportunity to work with him. His guidance, support, technical insights, and constructive criticism have helped me throughout my PhD research. His experience and expertise have helped me immensely in formulating and pursuing my dissertation topic.

I thank Prof. Shobha Bhatia for chairing the dissertation defense. I thank Prof. Chilukuri K. Mohan, Prof. Edmund S. Yu, Prof. Reza Zafarani and Prof. Sucheta Soundarajan for agreeing to be on my research committee. I am grateful to the chair and the committee for generously offering their time, support and goodwill.

I thank Prof. Sitharama S. Iyengar and Prof. Zhanpeng Jin for giving me the opportunity to collaborate with them and for their guidance. I am thankful to all the collaborators for providing me a great learning experience and helping me succeed in my PhD research.

I am grateful to my lab mates for all their help and for providing a healthy learning environment during my PhD. I thank all my friends for their constant support and encouragement.

I thank my family for their unconditional love and support.

## TABLE OF CONTENTS

	Page
ABSTRACT . . . . .	i
LIST OF TABLES . . . . .	xi
LIST OF FIGURES . . . . .	xiv
1 Introduction . . . . .	1
1.1 Overview of dissertation . . . . .	3
1.2 Key contributions of dissertation . . . . .	4
1.3 Published material in the dissertation . . . . .	5
2 Collecting and Sharing a Large Behavioral Biometric Dataset: Insights from BB-MAS . . . . .	7
2.1 Key contributions of the chapter . . . . .	8
2.2 Details of the data collection . . . . .	10
2.2.1 Devices used in data collection . . . . .	11
2.2.2 How was the data collected? . . . . .	13
2.2.3 What is the format of the data? . . . . .	15
2.2.4 Demographic details of the participants . . . . .	29
2.2.5 Where is it stored and how to obtain it? . . . . .	31
2.3 Initial analysis of keystroke data . . . . .	31
2.3.1 Statistics of keystroke data and insights from feature values across devices . . . . .	31
2.4 Possible research directions using the dataset . . . . .	33
2.5 Comparison with other data sets . . . . .	37
2.6 Lessons learnt from collection of the dataset . . . . .	40
2.7 Conclusion and future work . . . . .	42
3 Discriminative Power of Typing Features on Desktops, Tablets and Phones for User Identification . . . . .	44



	Page
3.1 Key contributions on the chapter . . . . .	46
3.2 Related Work . . . . .	47
3.3 Conventional keystroke features . . . . .	50
3.4 Proposed context sensitive features . . . . .	54
3.5 Details of data collection . . . . .	56
3.6 Feature discriminability analysis . . . . .	58
3.7 Analysis of conventional KD features . . . . .	59
3.7.1 KeyHold . . . . .	60
3.7.2 Analysis of conventional feature - Flight1 . . . . .	61
3.7.3 Analysis of conventional feature - Flight2 . . . . .	62
3.7.4 Analysis of conventional feature - Flight3 . . . . .	64
3.7.5 Analysis of conventional feature - Flight4 . . . . .	65
3.8 Analysis of proposed context sensitive features . . . . .	68
3.8.1 Analysis of proposed feature - WordHold . . . . .	69
3.8.2 Analysis of proposed feature - AvgFlight1 . . . . .	71
3.8.3 Analysis of proposed feature - AvgFlight2 . . . . .	72
3.8.4 Analysis of proposed feature - AvgFlight3 . . . . .	73
3.8.5 Analysis of proposed feature - AvgFlight4 . . . . .	75
3.8.6 Analysis of proposed feature - AvgKeyHold . . . . .	76
3.8.7 Analysis of proposed feature - StdFlight1 . . . . .	77
3.8.8 Analysis of proposed feature - StdFlight2 . . . . .	79
3.8.9 Analysis of proposed feature - StdFlight3 . . . . .	80
3.8.10 Analysis of proposed feature - StdFlight4 . . . . .	81
3.8.11 Analysis of proposed feature - StdKeyHold . . . . .	82
3.9 Evaluation of proposed features . . . . .	88
3.10 Insights drawn from the analysis on proposed features . . . . .	92
3.10.1 Insight 1: Why do the proposed features perform better than the conventional features? . . . . .	92

	Page	
3.10.2	Insight 2: Why are the results and the performance of proposed features better in case of hand-held devices when compared to the desktop? . . . . .	94
3.10.3	Insight 3: What word-based factors might impact the user identification performance of proposed features? . . . . .	96
3.10.4	Discussion: Attacks and limitations . . . . .	97
3.10.5	Conclusion and future work . . . . .	99
4	<b>Behavioral Biometrics : The Failure of Normality Assumption . . . . .</b>	<b>100</b>
4.1	Key contributions of the chapter . . . . .	101
4.2	Related work . . . . .	102
4.2.1	The Central Limit Theorem and Cràmer-Rao Lower Bound . . . . .	104
4.2.2	Related work in keystroke dynamics . . . . .	106
4.3	Data and features . . . . .	108
4.3.1	Details of the dataset . . . . .	108
4.3.2	Details of the features . . . . .	110
4.4	Experimentation and analysis . . . . .	112
4.5	Results and discussion . . . . .	114
4.6	Conclusion and alternate approaches . . . . .	115
5	<b>Classification of Threat Level in Typing Activity Through Keystroke Dynamics</b>	<b>123</b>
5.1	Key contributions of the chapter . . . . .	124
5.2	Related work . . . . .	125
5.3	Dataset and experimentation methods . . . . .	128
5.3.1	Details of the data collection . . . . .	128
5.3.2	Context recognition with conventional features . . . . .	130
5.3.3	Proposed features . . . . .	133
5.3.4	Context recognition with proposed features . . . . .	135
5.3.5	Correlation analysis of feature pairs . . . . .	137
5.4	Conclusion and future work . . . . .	137
6	<b>Exploratory work . . . . .</b>	<b>139</b>
6.1	Authentication by Mapping Keystrokes to Music: The Melody of Typing	139

	Page
6.1.1	Key contributions of the section . . . . . 140
6.1.2	Related work . . . . . 141
6.1.3	Details of the data collection . . . . . 143
6.1.4	Music features . . . . . 144
6.1.5	Analysis on music from keystrokes . . . . . 147
6.1.6	Inter-user and intra-user analysis . . . . . 148
6.1.7	Conclusion and future work . . . . . 154
6.2	DoubleType: Authentication Using Relationship Between Typing Behavior on Multiple Devices . . . . . 155
6.2.1	Key contributions of the section . . . . . 157
6.2.2	Related work . . . . . 157
6.2.3	Overview of the authentication system . . . . . 160
6.2.4	Details of the data collection . . . . . 162
6.2.5	Methodology and experiments . . . . . 163
6.2.6	Results and discussion . . . . . 168
6.2.7	Conclusion and future work . . . . . 169
7	Summary . . . . . 170
A	Additional Details of Data Collection . . . . . 173
A.1	Cognitive Loads[35] . . . . . 173
A.2	Examples of Free text questions on desktop . . . . . 173
A.3	Examples of Free text questions on tablet . . . . . 173
A.4	Examples of Free text Questions on phone . . . . . 174
A.5	Transcription Sentences . . . . . 174
	LIST OF REFERENCES . . . . . 175
	VITA . . . . . 192

## LIST OF TABLES

Table	Page
2.1 Data collection tasks performed by the participants. For each participant we recorded activities on four devices - a Desktop, a Tablet and two Phones (pocket and hand). . . . .	16
2.2 Example for keystroke files from user 1. . . . .	18
2.3 Example for mouse movement data from user 1. . . . .	19
2.4 Example for mouse button data from user 1. . . . .	19
2.5 Example for mouse wheel data from user 1. . . . .	19
2.6 Example for accelerometer data from user 1. . . . .	20
2.7 Example for gyroscope data from user 1. . . . .	20
2.8 Example for swipe data from user 1. . . . .	22
2.9 Example for checkpoint data from user 1. . . . .	23
2.10 Summary of demographic data. . . . .	30
2.11 Keystroke statistics: Number of keystroke events. . . . .	32
2.12 Summary of keyhold feature statistics. All values are in milliseconds. . . . .	34
2.13 Summary of Flight1 feature statistics. All values are in milliseconds. . . . .	34
2.14 Summary of Flight2 feature statistics. All values are in milliseconds. . . . .	35
2.15 Summary of Flight3 feature statistics. All values are in milliseconds. . . . .	36
2.16 Summary of Flight4 feature statistics. All values are in milliseconds. . . . .	37
2.17 Comparison with other related datasets. . . . .	38
3.1 Conventional features extracted from Uni-Graphs and Di-Graphs with their brief description. . . . .	52
3.2 Conventional features extracted from an example string "this is that". $U$ : Uni-Graph, $D$ : Di-Graph. $t_{1R}$ stands for release of key $t_1$ (the subscript 1 stands for the first occurrence of "t") and $t_{1P}$ stands for press of key $t_1$ and so on. . . . .	52
3.3 Proposed context sensitive features and their brief description. . . . .	53

Table	Page
3.4 Proposed features extracted from the same example string "this is that". $t_{1R}$ stands for release of key $t_1$ (the subscript 1 stands for the first occurrence of "t") and $t_{1P}$ stands for press of key $t_1$ and so on, Avg and Std stand for average and standard deviation respectively. . . . .	54
3.5 The Inter-User $Dist_B$ values for KeyHold distributions on all devices. . . . .	60
3.6 The Inter-User $Dist_B$ values Flight1 distributions on all devices. . . . .	61
3.7 The Inter-User $Dist_B$ values for Flight2 distributions on all devices. . . . .	63
3.8 The Inter-User $Dist_B$ values for Flight3 distributions on all devices. . . . .	64
3.9 The Inter-User $Dist_B$ values for Flight4 distributions on all devices. . . . .	66
3.10 The Inter-User $Dist_B$ values for WordHold Distributions across all devices.	70
3.11 The Inter-User $Dist_B$ for AvgFlight1 Distributions across all devices. . . . .	71
3.12 The Inter-User $Dist_b$ values for AvgFlight2 Distributions across all devices.	72
3.13 The Inter-User $Dist_B$ values for AvgFlight3 Distributions across all devices.	74
3.14 The Inter-User mean $Dist_B$ values for AvgFlight4 Distributions across all devices. . . . .	75
3.15 The Inter-User $Dist_B$ values for AvgKeyHold Distributions across all devices. . . . .	76
3.16 The Inter-User $Dist_B$ values for StdFlight1 Distributions across all devices.	78
3.17 The Inter-User $Dist_B$ values for StdFlight2 Distributions across all devices.	79
3.18 The Inter-User $Dist_B$ values for StdFlight3 Distributions across all devices.	80
3.19 The Inter-User $Dist_B$ values for StdFlight4 Distributions across all devices.	82
3.20 The Inter-User $Dist_B$ values for StdKeyHold Distributions across all devices.	83
3.21 Classifier accuracies for the conventional feature based classifiers in our experiment. . . . .	91
3.22 Classifier accuracies for the proposed feature based classifiers in our experiment. . . . .	91
3.23 Example from our desktop dataset: average feature values for a randomly chosen user shows the variations in the average feature values for the character "h" and digraph "ha" depending on the context in which they appear. All values are in milliseconds. . . . .	94
4.1 The different types of data, from SU-AIS BB-MAS [26], that we analyzed from multiple devices and activities. The gait activity consists of three sub-activities, walking, climbing upstairs and downstairs. . . . .	109

Table	Page
4.2 List of features extracted and examined in our experiments for an underlying normal distribution. . . . .	109
4.3 Percentage of test samples with $p > 0.05$ for keyhold feature from unigraphs on desktop, tablet and phone. . . . .	118
4.4 Percentage of test samples with $p > 0.05$ for flight1-flight4 features from digraphs on desktop. . . . .	118
4.5 Percentage of test samples with $p > 0.05$ for flight1-flight4 features from digraphs on tablet. . . . .	119
4.6 Percentage of test samples with $p > 0.05$ for flight1-flight4 features from digraphs on phone. . . . .	119
4.7 Percentage of test samples with $p > 0.05$ for features from Swiping activity on phone and tablet. . . . .	119
4.8 Percentage of test samples with $p > 0.05$ for features from Upstairs activity with phone in hand, phone in pocket and tablet in hand. . . . .	120
4.9 Percentage of test samples with $p > 0.05$ for features from Downstairs activity phone in hand, phone in pocket and tablet in hand. . . . .	121
4.10 Percentage of test samples with $p > 0.05$ for features from Walking activity phone in hand, phone in pocket and tablet in hand. . . . .	122
5.1 Highlights of our data collection effort. . . . .	126
6.1 The average FAR, FRR and Accuracy; for the three standard classifiers with two-fold and three-fold cross validation experiments (on the left) and for human classifiers (on the right) on user verification. . . . .	151
6.2 Summary of the data collection. . . . .	161

## LIST OF FIGURES

Figure	Page
2.1 A screenshot from our phone application keyboard which matches the default android keyboard. . . . .	12
2.2 The data collection procedure. Tasks <b>a</b> to <b>m</b> were performed by participants in sequence, the corresponding activities and data collected are described in Table 2.1 . . . . .	13
2.3 Organisation of the files in our dataset. . . . .	17
2.4 Features extracted from keystroke data. . . . .	25
3.1 Features extracted from the temporal data of keys $K_i$ and $K_{i+1}$ . . . . .	50
3.2 Highlights of our Data Collection effort. . . . .	56

Figure	Page
3.3 Example of $Dist_B$ computation: Histograms representing the probability density functions of KeyHold values for the character 't', for Users A and B on a) desktop , b) tablet and c) phone along with their corresponding Bhattacharyya distance. . . . .	59
3.4 Comparing the Bhattacharyya distances of PDFs for all conventional features on desktop, tablet and phone. . . . .	67
3.5 Comparing the Bhattacharyya distances of PDFs for all proposed context-sensitive features on desktop, tablet and phone. . . . .	85
3.6 Typically, a desktop keyboard offers only two degrees of freedom; forward/backward and left/right. As a typical phone can be held by its user in any comfortable posture, it offers six degrees of freedom; forward/backward, left/right, upward/downward, yaw, pitch and roll as shown in these figures. . . . .	96
3.7 Impact of three word-based factors on the performance of proposed features for user identification. The three factors are: Word length (Fig. 3.7a): number of characters in a word; Vowel Percentage (Fig. 3.7b): percentage of vowels in a word; and Oxford English Corpus (OEC) frequency ranking (Fig. 3.7c): the frequency ranking of the words in our study according to OEC (Top 100). The words in order of rank (Fig. 3.7c, y-axis) are: (1, the), (8, that), (9, have), (13, not), (15, with), (21, this), (33, will), (38, there), (69, see), (84, two) and (88, first). . . . .	96
4.1 Illustration summarizing the amount of features in each activity and percentage of their samples with $p > 0.05$ , or in other words, where the null hypothesis $H_0$ , that the samples came from a normal distribution could not be discarded. The categories and their corresponding color codes are; <b>red</b> - less than 25% of samples with p-value $>0.05$ ; <b>orange</b> - 25% to 50% of samples with p-value $>0.05$ ; <b>yellow</b> - 50% to 75% of samples with p-value $>0.05$ ; and, <b>green</b> -above 75% samples with p-value $>0.05$ . A full doughnut in the doughnut chart represents all the features for an activity on the labelled device. For example, the outer most doughnut in Figure 4.1a, represents all the features examined for keystrokes latencies on desktop, the second doughnut for tablet and innermost doughnut for phone respectively. The area covered by each color/category on a doughnut represents the amount of features that fall in the color/category as described above. . . . .	111
5.1 The accuracies, FPRs (Type 1 error) and FNRs (Type 2 error) from the SVM, RF and MLP classifiers trained and tested using the conventional keystroke features. . . . .	131
5.2 The accuracies, FPRs (Type 1 error) and FNRs (Type 2 error) from the SVM, RF and MLP classifiers trained and tested using our proposed keystroke features. . . . .	133



Figure	Page
5.3 Heat-maps showing the correlation between feature pairs in the proposed feature set for different sizes of text samples. $F_1$ to $F_{14}$ on the x-axis and y-axis represent the features <i>AvgEnterHold</i> , <i>StdEnterHold</i> , <i>AvgSpaceInFlight</i> , <i>StdSpaceInFlight</i> , <i>AvgSpaceOutFlight</i> , <i>StdSpaceOutFlight</i> , <i>SpaceRatio</i> , <i>EnterRatio</i> , <i>ErrorCount</i> , <i>TotalTime</i> , <i>IQRHold</i> , <i>IQRFlight</i> , <i>PunctuationRatio</i> and <i>SpeedDelta</i> , respectively . . . . .	133
5.4 The accuracies, FPRs (Type 1 error) and FNRs (Type 2 error) from the SVM, RF and MLP classifiers using eight least correlated features from our proposed set. . . . .	136
6.1 Music notes and their placement, generated from a digraph $K_i, K_{i+1}$ using functions $T(v)$ and $P(v)$ for duration and pitch respectively. . . . .	144
6.2 Examples of the piano roll plots that are obtained after mapping the keystroke features to the music features. We illustrate the piano roll plots of two test-phrase samples from two random users from our data-set, Figures 6.2(a) and 6.2(b) are from samples of user A and Figures 6.2(c) and 6.2(d) are from user B. . . . .	146
6.3 Plot of density functions for inter-user and intra-user Canberra distances of the note-pitch vectors (6.3a) and note-duration vectors (6.3b) between all music files. . . . .	149
6.4 Results from the Human-Classifier (HC) based verification experiments. . .	152
6.5 An overview of the authentication system. . . . .	158
6.6 Data preprocessing and formation of datasets from the relationship between typing behavior on two devices. . . . .	163
6.7 Illustration of the feature values from two random users selected from our dataset for scenario 1: <i>relationship – features</i> for Desktop and Phone. . .	163
6.8 Performance of the two classifiers for all three scenarios, Desktop-Phone (6.8a and 6.8b); Desktop-Tablet (6.8c and 6.8d); and Tablet-Phone (6.8e and 6.8f) relationship. . . . .	166

## 1. INTRODUCTION

The rise in the popularity of biometrics stems from its inherent property that eliminates the need for a person to remember and reproduce complex secretive information or carry additional hardware to authenticate oneself. Possession of such secretive information or hardware is not only risking their theft but also risking forgetting them (Personal Identification Number (PIN), password), either case leads to unnecessary complications regarding an individual's identity. Biometrics focuses on authenticating a person based on "who they are" rather than "what they know", which is a prime reason for its growth in popularity and research (eg. see [53] and [32]).

Behavioral biometrics is a branch of biometrics that focuses on using a person's behavior or way of doing a task as means of authentication. These tasks can be any common, day to day tasks like walking, sleeping, talking, typing and so on. As interactions with computers and other smart-devices like phones and tablets have become an essential part of modern life, a person's style of interaction with them can be used as a powerful means of behavioral biometrics. However, there is a lack of large datasets with multiple activities, such as typing, gait and swipe performed by the same person. Furthermore, large datasets with multiple activities performed on multiple devices by the same person are non-existent. The difficulties of procuring devices, participants, designing protocol, secure storage and on-field hindrances may have contributed to this scarcity. The availability of such a dataset is crucial to forward the research in behavioral biometrics as usage

of multiple devices by a person is common nowadays. We present our SU-AIS BB-MAS dataset, with a total of about: 3.5 million keystroke events; 57.1 million data-points for accelerometer and gyroscope each; 1.7 million data-points for swipes; and enables future research to explore previously unexplored directions in inter-device and inter-modality biometrics. A common assumption in behavioral biometrics, is that feature values follow a normal distribution. This assumption impacts key facets of research such as decisions of sampling techniques and authentication models and performance and results from the resulting systems. We question the assumption of normality in the features extracted from the data.

Typing is a common form of interaction, where a person provides input for these devices either on keyboards or touch screens, thus making research in Keystroke Dynamics (KD) popular. Research in KD has grown far and wide, Umphress and Williams [180], in their work, demonstrated that keystroke behavior on keyboards/typewriters was indeed a distinguishable trait among users while more recent research has shown that KD can also be used on other devices that involve typing, such as phones and tablets [47], [126]. A considerable amount of research has also explored the effects of the type of text used for KD, that is fixed text vs free text [4]. The problem of authenticating users by their typing behavior has also been addressed from multiple perspectives as far as the underlying algorithms are concerned. Although research in KD has been advancing rapidly, there have been very few attempts to understand the impact of context on the features that are used for KD.

A user can accomplish various tasks through keystroke inputs. Intuitively some activities are benign in nature while others are malicious. Common day-to-day activities like

writing emails, documents or browsing the internet can possess a lesser threat to the system from the user, whereas activities involving terminal commands may possess greater threats. System intrusion detection has been explored by many researchers [31, 150] who have proposed solutions at various levels of system interaction, ranging from system calls to data mining techniques [3, 189]. We explore the possibility of using typing behavior to detect malicious activity.

## **1.1 Overview of dissertation**

The dissertation is presented as follows. Chapter 1 introduces the thesis and provides an overview of the material presented within the dissertation. Chapter 2 describes the SU-AIS BB-MAS dataset that we collected and analysed. It also provides insights on collecting and sharing large behavioral biometric datasets. Chapter 3 presents the details of the context specific keystroke features that we designed and evaluated for user authentication. Chapter 4 presents analysis of underlying distribution of data from all modalities in our dataset and the experiments to examine the assumption of normality. Chapter 5 describes our experiments to differentiate benign typing activity from adversarial typing activity using a new keystroke feature set that we proposed. Chapter 6 presents two exploratory research directions that we explored using our SU-AIS BB-MAS dataset, (a) authentication of users through music notes generated by mapping their keystroke latencies to music and (b) authenticating users through the relationship between their keystroke latencies on multiple devices. The related work for chapters 2 - 6 is presented as separate sections in

each corresponding chapter. Chapter 7 summarizes the dissertation. Finally, Appendix A provides additional information of our data collection efforts.

## 1.2 Key contributions of dissertation

The key contributions of our work, detailed in this dissertation are listed below:

- **Develop context specific keystroke features:** **a)** We show the shortcomings of conventional keystroke features for user identification. **b)** We propose and evaluate a set of keystroke features that take advantage of the context from which the latencies are extracted. Our evaluations show high accuracies of user identification using proposed features on desktop, tablet, and phone. **c)** We draw insights and discuss impact factors that affect performance of user identification while using our proposed features.
- **Question the assumption of normality in behavioral biometrics data:** **a)** We question the common assumption in behavioral biometrics research that the data follows an underlying normal distribution. Experiments on our SU-AIS BB-MAS dataset show that the features extracted from gait, keystroke and swipes data do not follow a Gaussian distribution for all devices in our dataset. **b)** We discuss various approaches to handle non-normality in behavioral biometric data.
- **Share benchmark behavioral biometrics dataset:** We present details of our SU-AIS BB-MAS dataset, which is shared on IEEEDataport with open-access permissions. This dataset provides a unique advantage of having data from multiple

modalities (typing, gait, swiping) on multiple devices (desktop, tablet, phone) performed by the same person.

- **Detect threat level in typing activity through keystroke features:** a) We propose and evaluate keystroke features that have a mix of content and temporal information. b) Using proposed features we achieve high accuracies for classification of text samples into benign and adversarial categories.
- **Develop a method to map keystroke signature to musical signature:** We present a method to map keystroke features to derive the musical equivalent of a keystroke signature and is also extendable to other behavioral biometrics.
- **Explore multi-device typing behavior relationship:** We propose a set of features that relate the typing behavior of a person in multi-device environments. Our proposed features achieve high accuracies for user validation in all three scenarios of user's typing behavior relationships, a) desktop-phone; b) desktop-tablet; and c) tablet-phone.

### 1.3 Published material in the dissertation

The material presented within Chapter 3 was published as a peer-reviewed journal paper in the ACM Transactions on privacy and Security[24]. The material presented in Chapters 5 and 6 were published in peer-reviewed conference papers in the Proceedings of IEEE International conference on Artificial Intelligence and Signal Processing (AISP20) [23, 27, 28]. The material presented in Chapter 2 is currently under review as a peer-

reviewed journal paper for IEEE Transactions on Biometrics, Behavior, and Identity Science. The material presented in Chapter 4 is under submission as a peer-reviewed journal paper for IEEE Transactions on Information Forensics and Security.

The material published in AISP20 conference [23, 27, 28] will be extended to be published in peer-reviewed journals.

The dataset described in Chapter 2, is published on IEEEDataport [83]. It is listed as one of the most popular datasets on the portal (through IEEE emails to all members on 05/13/2020 and 07/21/2020) and has about 6000 views at the time of writing this dissertation.

## **2. COLLECTING AND SHARING A LARGE BEHAVIORAL BIOMETRIC DATASET: INSIGHTS FROM BB-MAS**

Behavioral biometrics are key components in continuous and active user authentication. Rigorous experimentation on large datasets is needed to develop state-of-the-art algorithms and draw meaningful insights. However, there is a lack of large datasets with multiple activities, such as typing, gait and swipe performed by the same person. Furthermore, large datasets with multiple activities performed on multiple devices by the same person are non-existent. The difficulties of procuring devices, participants, designing protocol, secure storage and on-field hindrances may have contributed to this scarcity. The availability of such a dataset is crucial to forward the research in behavioral biometrics as usage of multiple devices by a person is common nowadays.

Researchers have explored various modalities such as keystrokes ([20, 121, 158]), gait ([64, 65, 183]), swipes on touch screen ([59, 116, 157]) to name a few. With growing number of devices used by a person, research in continuous authentication or behavior analysis will have span across devices and activities to stay relevant. However, the scarcity of benchmark datasets for such scenarios are a hindrance. Several attempts have been made to provide benchmark datasets for a single activity like keystrokes ([12, 21, 58, 87, 88, 106, 173]), gait ([42, 63, 127, 193]) or swipe ([59, 62, 98, 157]) on a single device family like desktop or phone. Few attempts were also made to share benchmark datasets with multiple activities using single device ([11, 114]). However, no large bench-



mark dataset exists for multi-activity in multi-device scenario, where the same activities were performed by the same users on multiple devices. We attempt to fill this gap and provide a *benchmark dataset* with BB-MAS (Behavioral Biometrics Multi-device and multi-Activity data from Same users) dataset where the same participants have provided typing, gait and swiping data on desktop, phone and tablet.

A total of 117 participants voluntarily provided 3.5 million keystroke events; 57.1 million data-points for accelerometer and gyroscope each; 1.7 million data-points for swipes. Each participant performed typing (including transcription and free text), gait (including walking on a flat corridor, upstairs and downstairs) and swiping using desktop, phone, and tablet. The data collection spanned about 3 months and various anonymized demographics information is provided for each participant. The unique ID allocated to the participant is used on all devices and activities.

Key contributions follow.

## **2.1 Key contributions of the chapter**

- Provide this dataset as a benchmark resource to the community to compare performance for same user performing multiple activities over multiple devices for multiple modalities, such as typing, swiping, and gait. As of writing of this paper, this publicly available dataset has been accessed 5815 times (see <http://dx.doi.org/10.21227/rpaz-0h66>).
- To the best of our knowledge, a dataset with the typing, gait and touch data from the same users on desktop, tablet and phone is not available publicly at the time of

this writing. With data from 117 participants our dataset stands out as unique and rich for exploration in various directions. Each participant's session ranged between 2 to 2.5 hours, resulting in a total of about: 3.5 million keystroke events; 57.1 million data-points for accelerometer and gyroscope each; 1.7 million data-points for swipes.

- Describe, extract and share features that are commonly described in literature, for data from all activities alongside the raw data, thus providing a *ready-made* resource for researchers to compare their algorithms.
- Compare our dataset with other related datasets for keystroke, gait and swipe and highlight their novelty, differences, and advantages. Other datasets are limited in the variety of participants. We provide data for individuals from various age groups, gender, height, language and daily usage of desktop, phone, and tablet, and typing style.
- Provide insights on the distribution of keystroke feature values across desktop, tablet, and phone for the same user. We find the keyhold times are smaller in magnitude and inter-key latencies are larger in magnitude on hand-held devices when compared to desktop. We posit that, difference in number of fingers being in contact with the typing surface (fewer on hand-held device) may lead to such patterns.
- Discuss possible research directions using the BB-MAS dataset and share lessons learnt from this elaborate data-collection effort to help future researchers on similar endeavors.

Data collection was carried out between April and June of 2017, after the IRB approval from our university. All participants signed consent forms and have willingly participated in this data collection. All data has been anonymized and any personal identifiers in the data are removed. All subjects, their data and demographic information can only be referenced through the unique participant ID provided to them.

Although, we have posted the complete dataset on IEEE Dataport [83], this paper presents unique insights that are not available in the instructional ReadMe document.

In addition, the detailed description of data is interspersed with explanations of collection procedure, analysis and discussions of extracted features and data-snippets.

## **2.2 Details of the data collection**

The dataset was designed to capture the behavior of the same users performing various day-to-day activities, such as typing, gait and swipes on three commonly used devices such as, desktop, tablet, and phone. Activities were deliberately designed to mimic real-life scenarios, for instance, the typing activity consists of both fixed and free text data, the gait activity consists of walking on flat corridors, walking downstairs and upstairs and touch and swipe data consists of activities such as reading and scrolling. The raw data and the features extracted are shared publicly and can be accessed online at <http://dx.doi.org/10.21227/rpaz-0h66> [83] .

### 2.2.1 Devices used in data collection

Three most commonly used device types in current times were selected for our data collection. A desktop, tablet, and phone would cover most of our modern-day interactions with devices. The details of the devices used in our data collection are as below:

- **Desktops:** Two identical desktop stations were setup. Each desktop station consisted of a standard QWERTY keyboard (Dell kb212-b), an optical mouse (Dell ms111-p) and a Dell 21-inch monitor. The keystrokes, mouse movements and clicks were logged.
- **Tablets:** HTC-Nexus-9 tablets were used for the tablet section of the data collection. These tablets had a screen size of 8.9 inches, screen resolution of 1536 x 2048 pixels, device dimensions of 9 x 6 x 0.3 inches (Length X Width X Height) and weighed about 435 grams. Keystrokes, accelerometer, gyroscope, and touch were logged.
- **Phones:** Two different models of phones, Samsung-S6 and HTC-One phones were used in the data collection. The Samsung Galaxy S6 had a screen size of 5.1 inches and screen resolution of 1440 x 2560 pixels with body dimensions of 143.4 x 70.5 x 6.8 mm and weighing 138 grams, whereas the HTC-One had a screen size of 5.0 inches and screen resolution of 1080 x 1920 pixels with body dimensions of 146.4 x 70.6 x 9.4 mm and weighing 160 grams. Keystrokes, accelerometer, gyroscope, and touch were logged. The raw data files from different models are identified by the suffix in the file names explained in detail in Section 2.2.3.

As the default android keyboard does not allow logging of keystrokes, we created and used an android qwerty keyboard on screen which was similar to the default android qwerty keyboard. The phones and tablets were locked in portrait orientation and users were allowed to type on them with any comfortable posture that they preferred. The details of the data collected from these devices and their formats is described in Section 2.2.3. Figure 2.1 shows a screenshot of the application with the keyboard for phone. The application on tablet had the same layout but was scaled to match the default keyboard of an android tablet.

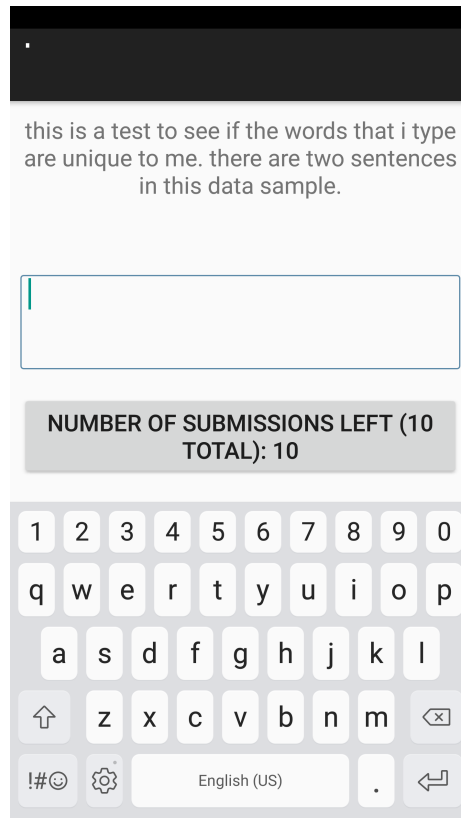


Fig. 2.1.: A screenshot from our phone application keyboard which matches the default android keyboard.

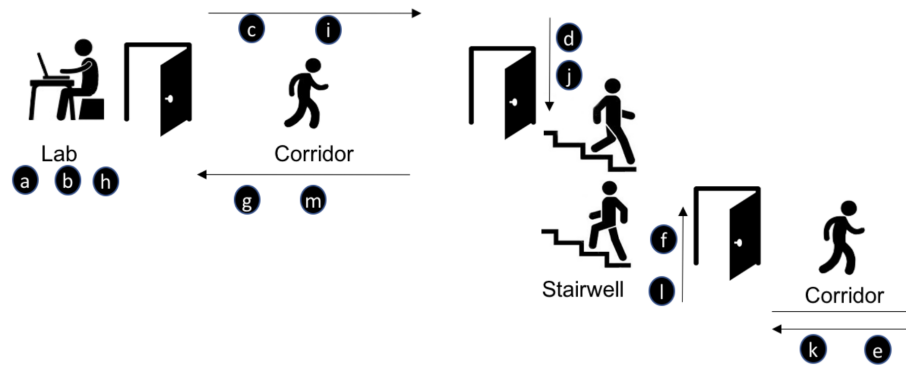


Fig. 2.2.: The data collection procedure. Tasks **a** to **m** were performed by participants in sequence, the corresponding activities and data collected are described in Table 2.1

### 2.2.2 How was the data collected?

Emails were sent out to all students, faculty, and staff to procure the participant population. Each participant had to spend two hours on average to perform the set of sequential tasks as illustrated in Fig. 2.2.

Upon arrival at the data collection location, each participant answered a set of questions that pertained to his/her demographics and technology usage. The participant was then assigned a unique ID and four devices: a desktop, a tablet and two phones (See Table 2.1). The participant then performed the tasks **a** to **m** in sequence. **a**) The participant was asked to sit at the desktop and type two sections of text (fixed-text), ten times each. Each piece of text consisted of two sentences and had an average of 112 characters. The participant was then given a shopping list consisting of six items. They had to use a popular web-browser (Mozilla Firefox) to browse for the best prices for the six items on the list while making notes (on any familiar text editor) about prices, opinions, and thoughts. The participant was then given a list of 12 questions of varying cognitive loads (see Appendix A.1 - A.3) and asked to type their answers in any order he/she preferred for roughly about

fifteen minutes. For the entire duration of task **a**, keystroke and mouse loggers were deployed on the desktop to log all the actions that the participant performed during this task.

**b)** After the completion of task **a**, the participant was handed a tablet which was running an application where he/she was asked to type the two pieces of static text again followed by a series of ten questions with varying cognitive loads to be answered with a minimum of 50 characters. The questions were placed in a manner that required the participant to swipe vertically and horizontally between questions. For the entire duration of task **b**, keystroke, touch, accelerometer, and gyroscope loggers were deployed on the tablet to log all typing, swiping, touch, and movement events. After the completion of task **b**, the participant was asked to place a phone (Phone1) in his/her pants pocket and made to walk in a predefined path while holding the tablet in hand. The path consisted of three doorways and a stairwell, as shown in Figure 2.2. The tablet displayed buttons to be pressed by the participant before and after passing through a doorway and also before and after taking the staircase. The tasks **c**, **e**, and **g** required the participant to walk, and tasks **d** and **f** required the participant to climb downstairs and upstairs respectively. Throughout the tasks **c** to **g**, the tablet and the phone (Phone1) logged the accelerometer and gyroscope values. The tablet also logged the pressing of the buttons (doorway and staircase) by the participant.

Upon completion of task **g**, the tablet was taken from the participant and another phone (Phone2) was handed to them. For task **h**, Phone2 ran the same application as the tablet in task **b**, where the participant had to type the two pieces of static text followed by a series of ten questions (not repeated from task **b**) with varying cognitive loads to be answered with a minimum of 50 characters, requiring the user to swipe between questions.

Phone2 logged all keystroke, touch, accelerometer and gyroscope values for typing, swiping, touch, and movement events. Tasks **i** to **m** are similar to tasks **c** to **g**, differing only in that the participant held Phone2 (instead of the tablet) and Phone1 remained in pocket while performing tasks **i** to **m**. Phone1 and Phone2 logged all accelerometer and gyroscope values. Phone2 also logged the pressing of buttons (doorway and staircase) by the participant. As the data collection involved logging of timestamps on multiple devices we made sure that clocks on all devices involved were synchronized to within a few milliseconds of each other by conducting several test runs to ensure synchronization.

### **2.2.3 What is the format of the data?**

The raw data from all sensors was originally written to sql databases for speed and accuracy. However, for the convenience of researchers, the raw data and the features extracted from them are organized in simple flat file structure in comma separated format (csv) shared at <http://dx.doi.org/10.21227/rpaz-0h66> [83]. This section elaborates the organization and format of both raw data files and feature extracted files. Fig. 2.3 gives an overview of the entire dataset. It is important that the dataset is clearly understood by its researchers for successful research. Therefore, we explain our dataset in great detail in this section.

#### **Description of the raw data**

The raw data from each sensor for each user is stored in folder labelled with the user's ID. As shown in Fig. 2.3, folders "1" to "117" contain the raw data files for each user, the



Table 2.1: Data collection tasks performed by the participants. For each participant we recorded activities on four devices - a Desktop, a Tablet and two Phones (pocket and hand).

Task	Device	Activity	Data	Duration (Approx.)
a	Desktop	Typing, Browsing	Keystroke and Mouse	50 min
b	Tablet	Typing	Keystroke, Swipe, Accelerometer, Gyroscope	25 min
c		Walking		
d	Tablet (in hand)	Climbing down stairs		
e	Phone1 (in pocket)	Walking	Accelerometer, Gyroscope	5 min
f		Climbing upstairs		
g		Walking		
h	Phone2 (in hand)	Typing	Keystroke, Swipe, Accelerometer, Gyroscope	25 min
i		Walking		
j	Phone2 (in hand)	Climbing down stairs		
k	Phone1 (in pocket)	Walking	Accelerometer, Gyroscope	5 min
l		Climbing upstairs		
m		Walking		

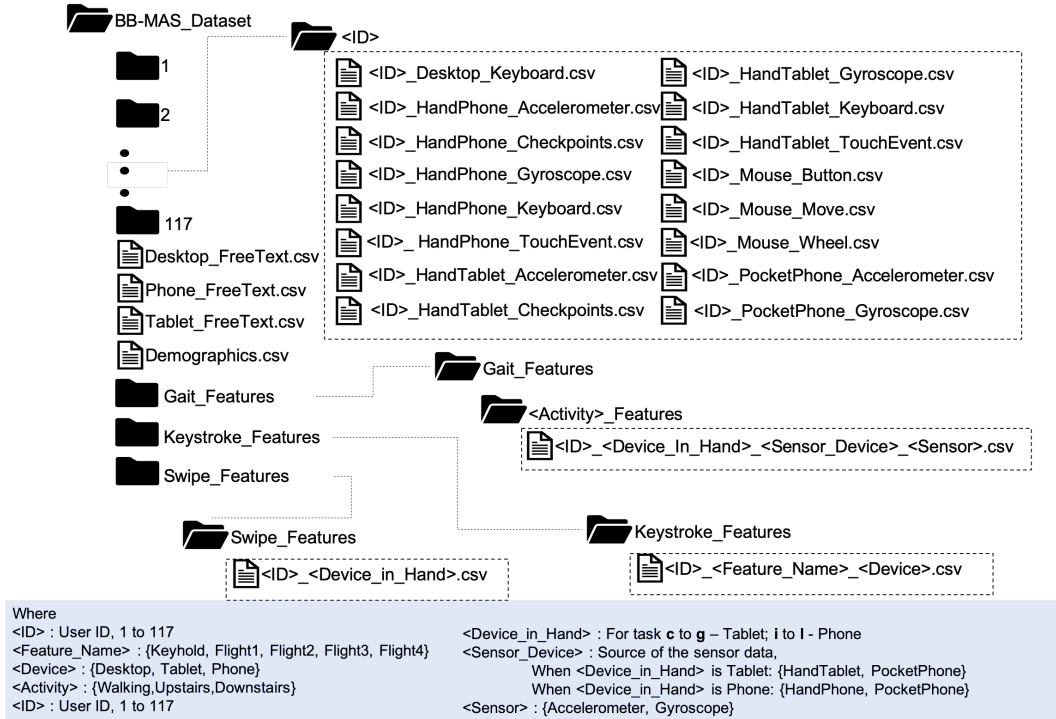


Fig. 2.3.: Organisation of the files in our dataset.

prefix `<ID>` is used to denote the user's ID. The details and format of the raw data files are as follows:

- **Keystroke Data:** The temporal data of every key press and release performed by the subject during tasks **a**, **b** and **h** (Table 2.1) were logged. These files are named;

- `<ID>_Desktop_Keyboard.csv`
- `<ID>_HandTablet_Keyboard.csv`
- `<ID>_HandPhone_Keyboard.csv`

accordingly. These files consist four columns, "EID": event ID (Integer); "key": the key triggering the key-event (String); "direction": the type of key-event (Integer, 0 for press and 1 for release); and "time": the timestamp of the key-event (String in date-

Table 2.2: Example for keystroke files from user 1.

EID	key	direction	time
0	t	0	2017-04-14 18:09:41.538
1	t	1	2017-04-14 18:09:41.679
2	i	0	2017-04-14 18:09:41.819
..	..	..	..

time format with millisecond resolution). Table 2.2 provides an example of keystroke files with a snippet from user 1 in our dataset.

- **Mouse Data:** In addition to keystrokes, data from mouse usage was also collected during task **a** (Table 2.1). Please note that there were sampling issues with the mouse data resulting in smaller files, they are included, nonetheless. Mouse events such as, movement, button and wheel were logged into files named;

- <ID>\_Mouse\_Move.csv

- <ID>\_Mouse\_Button.csv

- <ID>\_Mouse\_Wheel.csv

respectively. The Mouse\_Move file has six columns, "EID": event ID (Integer); "rX" and "rY": the x and y coordinates relative to the active window (Integer); "pX" and "pY": the x and y coordinate on screen (Integer); and "time": the timestamp of the mouse-event (String in date-time format with millisecond resolution). The Mouse\_Button file has eight columns, six of them are the same as described for Mouse\_Move, "LR": mouse button (Integer, 0 for left or 1 for right) and "state": type of button event (Integer, 0 for press and 1 for release) are the additional columns. The Mouse\_Wheel file has seven columns, six of them are the same as described for Mouse\_Move in addition

to, "delta": direction of scroll (Integer, Negative for scroll-down and positive for scroll-up). Tables 2.3, 2.4 and 2.5 provide an example mouse movement, button and wheel data respectively, from user 1.

Table 2.3: Example for mouse movement data from user 1.

EID	rX	rY	pX	pY	time
0	4	-8	1004	577	2017-04-14 18:09:29.948
1	8	-14	1919	0	2017-04-14 18:09:30.228
2	-2	-26	1916	0	2017-04-14 18:21:13.712
..	..	..	..	..	..

Table 2.4: Example for mouse button data from user 1.

EID	rX	rY	pX	pY	LR	state	time
0	6	-4	1285	242	0	0	2017-04-14 18:21:17.783
1	-1	3	811	265	0	1	2017-04-14 18:21:21.761
2	0	0	811	265	0	0	2017-04-14 18:21:22.120
..	..	..	..	..	..	..	..

Table 2.5: Example for mouse wheel data from user 1.

EID	rX	rY	pX	pY	delta	time
0	0	0	1594	708	120	2017-04-14 18:23:10.936
1	0	0	1545	708	120	2017-04-14 18:23:12.000
2	0	0	1618	708	120	2017-04-14 18:23:12.575
..	..	..	..	..	..	..

- **Accelerometer and Gyroscope Data:** For tasks from **b** through **m** (Table 2.1), the values from accelerometer and gyroscope sensors were logged on suitable devices, such as tablet: for tasks **c - g**; phone in pocket: for tasks **c - g** and **i - m**; and phone in hand: for tasks **i - g**. The sampling rate for these sensors was about 100Hz. The files with accelerometer and gyroscope from the tablet are named;

- <ID>\_HandTablet\_Accelerometer.csv

- <ID>\_HandTablet\_Gyroscope.csv

those from the phone in the pocket are named;

- <ID>\_PocketPhone\_Accelerometer.csv

- <ID>\_PocketPhone\_Gyroscope.csv

and from the phone in hand are named;

- <ID>\_HandPhone\_Accelerometer.csv

- <ID>\_HandPhone\_Gyroscope.csv

respectively. The accelerometer files have five columns, "EID": event ID (Integer); "Xvalue", "Yvalue", "Zvalue": the acceleration force in  $m/s^2$  on x, y and z axes respectively, excluding the force of gravity (Float); and "time": the timestamp of the data point (String in date-time format with millisecond resolution). The gyroscope data have the same five columns, but "Xvalue", "Yvalue" and "Zvalue" is the rate of rotation in rad/s around x, y and z axis respectively (Float). Tables 2.6 and 2.7 show an example for accelerometer and gyroscope data respectively, from user 1.

Table 2.6: Example for accelerometer data from user 1.

EID	Xvalue	Yvalue	Zvalue	time
0	1.043	3.245	9.087	2017-04-14 18:56:40.215
1	0.995	3.303	8.936	2017-04-14 18:56:40.216
2	0.988	3.355	8.880	2017-04-14 18:56:40.234
..	..	..	..	..

Table 2.7: Example for gyroscope data from user 1.

EID	Xvalue	Yvalue	Zvalue	time
0	-0.045	0.036	-0.013	2017-04-14 18:56:40.440
1	-0.027	0.027	-0.017	2017-04-14 18:56:40.449
2	-0.013	0.022	-0.017	2017-04-14 18:56:40.461
..	..	..	..	..

- **Swipe Data:** For tasks **b** and **h**, data from swipes were recorded on the tablet and phone in hand respectively. These were logged into files named;

- <ID>\_HandTablet\_TouchEvent.csv

- <ID>\_HandPhone\_TouchEvent.csv

respectively. The touch data files have ten columns, "EID": event ID (Integer); "Xvalue" and "Yvalue": the x and y coordinates on screen (Float), "pressure": the approximate pressure applied to the surface by a finger (Float, normalized to a range from 0 (no pressure at all) to 1 (normal pressure)), "touchMajor" and "touchMinor": the length of the major and minor axis, respectively, of an ellipse that represents the touch area (Float, display pixels), "pointerID": index of the pointer/touch used in case of multiple touch points (Integer), "fingerOrientation": the orientation of the finger in radians relative to the vertical plane of the device (Float, 0 radians indicates that the major axis oriented upwards, is perfectly circular or is of unknown orientation), "actionType": indicates the type of event (Integer, 0: finger down/swipe begin, 1: finger up/swipe end and 2: finger move/swipe); and "time": the timestamp of the data point (String in date-time format with millisecond resolution).

- **Checkpoints Data:** For the tasks **c - g** and **i - m**, we require checkpoints to separate the data into walking, upstairs and downstairs. The participants were asked to click on buttons on tablet (**c - g**) or phone in hand (**i - m**) to mark the opening and closing of doors and start and end of stairs. These checkpoints can be used to separate the data from all other sensors into different activities. Please note that a proctor followed the users during these tasks (making sure not to influence the activity) and noted down

Table 2.8: Example for swipe data from user 1.

ID	Xvalue	Yvalue	pressure	touchMajor	touchMinor	pointerID	fingerOrientation	actionType	time
0	818.085	1546.25	1.0	0.976	0.488	0	0.0	0.0	2017-4-14 18:56:47:185
1	819.140	1545.0	1.0	2.929	2.929	0	0.0	2.0	null
2	820.074	1543.893	1.0	2.929	2.929	0	0.0	2.0	2017-4-14 18:56:47:209

incidents where some users clicked the buttons either early or late by a few seconds, adjustments to such timestamps were made manually by adding or subtracting the number of seconds noted down by the proctor. Checkpoint files have three columns, "EID": event ID (Integer); "eventType": type of event (String, *DoorEntry*: user at doorway and is about to open door, *DoorExit*: user has crossed the doorway and the door has closed behind them, *StairEntry*: user about to start climbing up or down the staircase and *StairExit*: user has completed climbing up or down a staircase.); and "time": the timestamp of the data point (String in date-time format with millisecond resolution). Table 2.9 shows an example of checkpoint data from user 1. Using the checkpoint data, the accelerometer and gyroscope data can be segmented into three; a) between "DoorExit" and "DoorEntry" event represents walking on a flat corridor; b) between the first "StairEntry" and "StairExit" represents going downstairs; and between second "StairEntry and "StairExit" represents going upstairs.

Table 2.9: Example for checkpoint data from user 1.

EID	eventType	time
0	DoorEntry	2017-04-14 19:41:45.980
1	DoorExit	2017-04-14 19:41:50.639
..	..	..
4	StairEntry	2017-04-14 19:42:18.724
5	StairExit	2017-04-14 19:42:39.105
..	..	..

- **FreeText Data:** In tasks **a**, **b** and **h**, the users had to first transcribe two pieces of fixed text; a) "this is a test to see if the words that i type are unique to me. there are two sentences in this data sample."<sup>1</sup>; and b) "second session will have different set of lines.

<sup>1</sup>The transcription sentences were selected based on two criteria: (1) inclusion of many frequently used words in the Oxford English Corpus, and (2) encouraging typing activity on both hands (on both sides on the keyboard). Transcription sentences were typed in lower case.



carefully selected not to overlap with the first collection phase.”<sup>1</sup>. The files labelled ”Desktop\_FreeText.csv”, ”Tablet\_FreeText.csv” and ”Phone\_FreeText.csv” provide the timestamp for each user, at which they completed the transcription section and moved to free text section. In our analyses we have considered the entire typing activity as a whole, these files are provided for researchers who may want to separate fixed text and free text for their work.

### Features from raw data

We extracted popular features that are used in literature for each modality. The feature extraction for our dataset can be grouped into three parts, namely keystroke, gait and swipe features. The files consisting the extracted features have also been included in our dataset. We briefly describe the features and their storage below.

- **Keystroke Features:** We select the common twelve unigraphs (single key) and eighteen digraphs (pair of consecutive keys) that occurred the most number of times in all user’s keystroke data. The unigraphs are: ”BACKSPACE”, ”SPACE”, ”a”, ”e”, ”h”, ”i”, ”l”, ”n”, ”r”, ”S” and ”t”. The digraphs are: (’BACKSPACE’, ’BACKSPACE’), (’SPACE’, ’a’), (’SPACE’, ’i’), (’SPACE’, ’s’), (’SPACE’, ’t’), (’e’, ’SPACE’), (’e’, ’n’), (’e’, ’r’), (’e’, ’s’), (’n’, ’SPACE’), (’o’, ’SPACE’), (’o’, ’n’), (’r’, ’e’), (’s’, ’SPACE’), (’s’, ’e’), (’t’, ’SPACE’), (’t’, ’e’) and (’t’, ’h’). For a unigraph  $K_i$  we extract the *Keyhold* time of the key as a feature:

$$- \text{Keyhold}_{K_i} : K_i \text{Release} - K_i \text{Press}$$

For a digraph  $K_i$  and  $K_{i+1}$  the following temporal features are extracted:

- $Flight1_{K_i K_{i+1}} : K_{i+1} Press - K_i Release$
- $Flight2_{K_i K_{i+1}} : K_{i+1} Release - K_i Release$
- $Flight3_{K_i K_{i+1}} : K_{i+1} Press - K_i Press$
- $Flight4_{K_i K_{i+1}} : K_{i+1} Release - K_i Press$

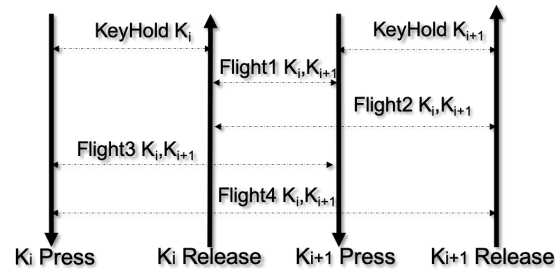


Fig. 2.4.: Features extracted from keystroke data.

The figure 2.4 illustrates the temporal features extracted from keystrokes. These files are stored in folder labelled "Keystroke\_Features" which contains the files names with syntax " $\langle ID \rangle\_ \langle Feature\_Name \rangle\_ \langle Device \rangle.csv$ ", where,  $\langle ID \rangle$  is the user ID (1-117),  $\langle Feature\_Name \rangle$  is the keystroke feature (keyhold, flight1 - flight4) and  $\langle Device \rangle$  is either desktop, tablet or phone. Each of these files have column denoting the key ("key" in case of keyhold, "key1" and "key2" in the case of flight) and a column with the value extracted for the feature.

- **Gait Features:** As the raw data for the gait is a pair of signals from the accelerometer and gyroscope we extract features from both. The gait data is further subdivided into three activities; "Walking" (on a flat corridor); "Downstairs" (going down the staircase); and "Upstairs" (going up the staircase). We use a window size of two seconds with a one second overlap between two consecutive windows. For each two second

window we extract a host of features from the accelerometer and the gyroscope for x ("Xvalue"), y ("Yvalue"), z ("Zvalue") and m ( $m = \sqrt{x^2 + y^2 + z^2}$ ). A brief description of the features and their column names in files are as follows:

- Mean: mean of x, y, z and m data denoted *xmean*, *ymean*, *zmean* and *mmean* respectively.
- Standard deviation: standard deviation of x, y, z and m data denoted *xstd*, *ystd*, *zstd* and *mstd* respectively.
- Band power: band power x, y, z and m data denoted *xbp*, *ybp*, *zbp* and *mbp* respectively.
- Energy: energy of the signals x, y, z and m denoted *xenergy*, *yenergy*, *zenergy* and *menergy* respectively.
- Median frequency: median frequency of x, y, z and m signals denoted *xmfreq*, *ymfreq*, *zmfreq* and *mmfreq* respectively.
- Inter quartile range: the inter quartile range of x, y, z and m data denoted *xiqr*, *yiqr*, *zigr* and *miqr* respectively.
- Range: range of the x, y, m and z signals denoted *xrange*, *yrange*, *zrange* and *mrangle* respectively.
- Signal to noise ratio: the signal to noise ratio in x, y, z and m signals denoted *xsnr*, *ysnr*, *zsnr* and *msnr* respectively.
- Dynamic time warping distance: the DTW distance between pairs of signals x-y, y-z and x-z denoted as *xydtw*, *yzdtw* and *xzdtw* respectively.

- Mutual information: the mutual information between pairs of signals x-y, x-z, x-m, y-z, y-m and z-m denoted as  $xymi$ ,  $xzmi$ ,  $xmmi$ ,  $yzmi$ ,  $ymmi$  and  $zmmi$  respectively.
- Correlation: the Pearson correlation coefficients between pairs of signals x-y, y-z and x-z signals denoted  $xycorr$ ,  $yzcorr$  and  $xzcorr$  respectively.

In the "Gait\_Features" folder, we have sub-folders named "<Activity>\_Features" where activity is either Walking, Downstairs or Upstairs. Each folder consists files with names following the syntax "<ID>\_<Device\_In\_Hand>\_<Sensor\_Device>\_<Sensor>.csv", where, <ID> is the user ID (1-117), <Device\_In\_Hand> is "Tablet" for tasks **c - g** and Phone for tasks **i - l**, <Sensor\_Device> is the device from which data comes from (HandPhone, HandTablet, PocketPhone) and <Sensor> is either accelerometer or gyroscope.

- **Swipe Features:** For each swipe performed by users on tablet and phone during tasks **b** and **h** respectively, various features related to the speed and trajectory of the swipes are extracted. A brief description of the features and their column names in files are as follows:

- Minimum x and y coordinates: the minimum x and y coordinates in the entire swipe denoted by  $minx$  and  $miny$  respectively.
- Maximum x and y coordinates: the maximum x and y coordinates in the entire swipe denoted by  $maxx$  and  $maxy$  respectively.

- Euclidean distance: the Euclidean distance between the start and end points of the swipe denoted by *eucliddist*.
- Distance list: Euclidean distance between points of a swipe denoted by *dlist*.
- Angle: the tangent angle of the swipe denoted by *tanangle*.
- Time: the total time taken to for the swipe denoted by *tottime*.
- Velocity mean and standard deviation: the mean and standard deviation of velocity during the swipe, *vmean* and *vstd* respectively.
- Velocity quartiles: the first, second and third quartiles of velocity during the swipe, *vquarts\_0*, *vquarts\_1* and *vquarts\_2* respectively.
- Acceleration mean and standard deviation: the mean and standard deviation of acceleration during the swipe, *amean* and *astd* respectively.
- Acceleration quartiles: the first, second and third quartiles of acceleration during the swipe, *aquarts\_0*, *aquarts\_1* and *aquarts\_2* respectively.
- Pressure mean and standard deviation: the mean and standard deviation of pressure during the swipe, *pmean* and *pstd* respectively.
- Pressure quartiles: the first, second and third quartiles of pressure during the swipe, *pquarts\_0*, *pquarts\_1* and *pquarts\_2* respectively.
- Area mean and standard deviation: the mean and standard deviation of area during the swipe, *areamean* and *areastd* respectively.
- Area quartiles: *areaquarts\_0*, *areaquarts\_1* and *areaquarts\_2* respectively.

- Direction: the direction of the swipe comparing the displacement of the fingertip in x and y direction, the direction of swipes is deduced as either left, right, up or down denoted by column *swipe\_type*.

These files are stored in the directory "Swipe\_Features" and named with syntax "<ID>\_<Device\_in\_Hand>.csv", where, <ID> is the user ID (1-117) and <Device\_in\_Hand> (Tablet or Phone) is the device on which the swipe was performed.

#### **2.2.4 Demographic details of the participants**

Each participant was given a unique ID and made to fill out a brief questionnaire consisting questions relating to demographic, physiology and background. This data is stored in the file labelled "Demographics.csv" in the form of thirteen columns, "User ID": unique ID given to each user (Integer); "Age": age of the participant in years (Integer); "Gender": the gender of the participants (Character, "F": Female, "M": Male and "O": Other); "Height": height of the participant in inches (Integer, inches); "Ethnicity": ethnicity of the participant (String); "Languages Spoken": languages that the participant can speak fluently (Tuple, [language1, ..., languageN]); "Typing Languages": languages in which the participant can type (Tuple, [language1, ..., languageN]); "Handedness": dominant hand for the participant ("Right", "Left" or "Ambidextrous"); "Desktop Hours", "Smartphone Hours" and "Tablet Hours": approximate range of hours in a day the participant spends using a desktop, phone and tablet respectively (Range, in hours: 0-1, 2-4, 5-7, 8-12, More than 12); "Typing Style": denotes touch and visual typists (Character, "a": Do not look at keyboard/Touch typist, "b": Must look at keyboard/Visual typist and "c": Occasionally

look at keyboard/Visual typist); and "Major/Minor": participant's major and minor stream of education (String).

Table 2.10 summarizes the demographics of the participants in our dataset. The average age of participants in our study was about 25 years with more than half of the participants aged between 23 to 26 years. The youngest and oldest participants were 19 and 35 years of age respectively. The average height of participants was about 67 inches. The shortest and tallest being 54 and 74 inches respectively. The daily usage hours also reflect the popularity of these devices while desktops and phones appear to be used more than tablets.

Table 2.10: Summary of demographic data.

Category	Size	Category	Size		
Age in years	19 - 22	22	Daily usage of desktop in hours	0 - 1	17
	23 - 26	61		2 - 4	58
	27 - 30	28		5 - 7	28
	>30	06		8 - 12	12
Sex	Female	45	>12	2	
	Male	72	Daily usage of phone in hours	0-1	3
Height in inches	≤60	6		2-4	51
	60-65	40		5-7	43
	65-70	43		8-12	16
	>70	28	>12	4	
Spoken Languages	1	13	Daily usage of tablet	0 - 1	93
	2	64		2-4	20
	3	32		5-7	4
	>3	8	Typing style	Touch	31
			Visual	86	

### **2.2.5 Where is it stored and how to obtain it?**

The entire dataset, feature files and demographic file are hosted at IEEE-Dataport [83].

The url of the dataset is <http://dx.doi.org/10.21227/rpaz-0h66> and is open-access complaint, so it can be downloaded with a free IEEE account.

## **2.3 Initial analysis of keystroke data**

Through this paper, we present results from initial research directions that we have explored along with the in-dept description of our dataset. We hope the research community will benefit from the dataset and explore the various other directions of research that cannot be addressed in one single research article. We collect the statistics of the keystroke data and compare the average and standard deviation of the keystroke features (Section 2.2.3) across the three different devices. This helps us provide insights about general typing behavior traits on various devices.

### **2.3.1 Statistics of keystroke data and insights from feature values across devices**

Table 2.11 shows the statistics of keystroke data from our dataset. On an average each participant performed around 11,750, 8,950 and 9,400 keystrokes on desktop, tablet and phone respectively. Even in the minimum condition, each participant has performed 4,350, 4,550 and 5,450 keystrokes on the three devices respectively. When combined, the average keystrokes per user across all three devices is over 30,000 keystrokes and at minimum about 19,250.



Table 2.11: Keystroke statistics: Number of keystroke events.

	Desktop	Tablet	Phone	All Devices
Average	11760	8952	9395	30153
Stdev	2132	1584	1472	3880
Min	4365	4580	5463	19252
Max	18716	17029	14694	41828

**Outlier Detection for Keystroke Features:** From the keystroke data, we extract the all temporal keystroke features that are popular in literature (See Section 2.2.3). We use a simple filter to remove any instances of keys that were held down for two seconds or more. We also remove instances of the inter-key pauses that are greater than two seconds. We assume that these were caused by pauses, where the user is either thinking or receiving instructions during the data collection.

### Insights from keystroke feature values across devices

To observe how the keystroke features vary across devices, we calculate the average of the average feature values and average of the standard deviation of the feature values from all users in our dataset. To maintain clarity in presentation we use symbols  $d1$  through  $d18$  to denote the digraphs ( $d1$ : ('BACKSPACE', 'BACKSPACE'),  $d2$ : ('SPACE', 'a'),  $d3$ : ('SPACE', 'i'),  $d4$ : ('SPACE', 's'),  $d5$ : ('SPACE', 't'),  $d6$ : ('e', 'SPACE'),  $d7$ : ('e', 'n'),  $d8$ : ('e', 'r'),  $d9$ : ('e', 's'),  $d10$ : ('n', 'SPACE'),  $d11$ : ('o', 'SPACE'),  $d12$ : ('o', 'n'),  $d13$ : ('r', 'e'),  $d14$ : ('s', 'SPACE'),  $d15$ : ('s', 'e'),  $d16$ : ('t', 'SPACE'),  $d17$ : ('t', 'e') and  $d18$ : ('t', 'h')). Tables 2.12 to 2.16 present the values computed for each feature.

**Observations:** We observe that participants have considerably less keyhold times for all keys on the hand-held devices (tablet and phone). The average of the standard deviation in keyhold time are also very small, less than 50 milliseconds, in all cases except for "backspace". However, it is almost completely the opposite when we consider the flight1 to flight4 features from Tables 2.13 to 2.16. We observe that in case of hand-held devices the average feature values are larger than those on desktop. Especially in case of flight1, which is also called the inter-key latency, we can see the values are almost doubled for many digraphs (*d6*, *d10*, *d11* etc.). It is also worth noticing that both the hand-held devices exhibit similar values in most cases and contrasts, if present, are only with the desktop keystroke features.

**Insights:** From our observations it appears that participants in general, take longer time between keys on phones and tablets when compared to a desktop keyboard. However, once the key is pressed the release event occurs much sooner on the phones and tablets implying that smaller amount of time is spent with the finger on the key. We posit that this occurrence maybe a result of lesser number of fingers being in contact with the typing surface on hand-held devices. In most cases participants type on tablets and phones with just their thumbs compared to their usage of many more fingers for the desktop keyboard thus increasing the keyhold time and reducing the inter-key latency on desktop.

## 2.4 Possible research directions using the dataset

We briefly discuss various research directions that can be explored with the help of our BB-MAS dataset below.

Table 2.12: Summary of keyhold feature statistics. All values are in milliseconds.

Unigraph	Desktop		Tablet		Phone	
	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$
bspace	168	211	128	175	128	129
space	114	57	77	18	89	17
a	137	68	98	22	103	19
e	123	58	85	20	90	18
h	116	53	73	16	81	16
i	119	61	71	16	85	17
l	102	50	71	16	89	19
n	122	63	72	16	83	16
o	118	61	72	15	87	16
r	129	63	78	19	85	17
s	130	60	87	21	94	18
t	116	54	76	20	81	16

Table 2.13: Summary of Flight1 feature statistics. All values are in milliseconds.

Digraph	Desktop		Tablet		Phone	
	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$
<i>d1</i>	20	258	80	280	20	228
<i>d2</i>	205	247	412	318	360	313
<i>d3</i>	277	247	513	316	472	290
<i>d4</i>	224	245	432	307	433	321
<i>d5</i>	218	240	441	327	408	298
<i>d6</i>	96	167	199	194	162	170
<i>d7</i>	115	151	175	138	144	126
<i>d8</i>	37	110	123	77	130	63
<i>d9</i>	123	114	169	87	162	80
<i>d10</i>	95	137	200	161	186	142
<i>d11</i>	95	110	254	151	236	120
<i>d12</i>	99	110	205	94	181	75
<i>d13</i>	22	92	118	73	121	59
<i>d14</i>	99	163	208	230	163	169
<i>d15</i>	87	97	148	67	147	63
<i>d16</i>	111	158	250	219	198	154
<i>d17</i>	58	94	138	81	136	68
<i>d18</i>	63	88	111	79	121	74

- **User authentication for individual devices using keystrokes, gait or swipes:** Our dataset provides multiple modalities, activities and scenarios which can be used separately as individual device or activities for user authentication data.

Table 2.14: Summary of Flight2 feature statistics. All values are in milliseconds.

Digraph	Desktop		Tablet		Phone	
	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$
<i>d1</i>	166	169	163	154	173	118
<i>d2</i>	333	248	509	310	462	308
<i>d3</i>	387	246	582	307	556	287
<i>d4</i>	348	247	520	301	529	317
<i>d5</i>	327	238	518	321	488	295
<i>d6</i>	204	165	278	192	253	170
<i>d7</i>	220	156	250	137	231	126
<i>d8</i>	174	114	200	80	217	65
<i>d9</i>	247	119	255	87	253	82
<i>d10</i>	212	148	275	157	274	144
<i>d11</i>	206	117	333	151	326	121
<i>d12</i>	228	122	275	94	263	76
<i>d13</i>	155	93	203	77	209	62
<i>d14</i>	209	165	286	225	255	170
<i>d15</i>	208	103	230	69	238	65
<i>d16</i>	219	158	328	215	288	154
<i>d17</i>	181	101	225	82	224	69
<i>d18</i>	182	97	186	79	202	77

- Activity recognition:** We share data from multiple activities and sub-activities like; free text and fixed text in case of keystrokes; and walking, upstairs and downstairs in case of gait. Recognizing the activity or sub-activity provides better context for methods to be applied.
- Feature engineering:** Many authentication and identification tasks can be improved with a better understanding of feature sets and their effectiveness for each of the modalities.
- Inter-Device behavior patterns:** A unique property of our dataset that sets it apart from openly available datasets is that, the same participants performed many overlapping activities on multiple devices. Therefore, inter-device patterns in behavior

Table 2.15: Summary of Flight3 feature statistics. All values are in milliseconds.

Digraph	Desktop		Tablet		Phone	
	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$
<i>d1</i>	194	156	179	146	187	110
<i>d2</i>	314	245	491	311	450	306
<i>d3</i>	387	248	588	307	559	288
<i>d4</i>	337	245	509	300	524	315
<i>d5</i>	326	236	518	320	496	294
<i>d6</i>	219	167	284	190	252	168
<i>d7</i>	238	154	265	136	236	120
<i>d8</i>	179	112	218	77	230	64
<i>d9</i>	246	122	251	86	248	80
<i>d10</i>	220	145	267	157	266	143
<i>d11</i>	210	114	326	152	320	119
<i>d12</i>	235	121	279	94	268	76
<i>d13</i>	163	93	197	74	208	60
<i>d14</i>	229	162	297	224	258	167
<i>d15</i>	219	100	237	69	246	63
<i>d16</i>	229	157	324	213	277	151
<i>d17</i>	178	93	216	81	220	69
<i>d18</i>	177	90	189	79	203	74

in same activity or different activities can be researched. For example, "*Can the typing behavior of a user on desktop reveal their typing behavior on phone?*".

- **Physiological or Demographic information leakage in activities:** As we provide a demographic information of each participant, researchers can also explore if the membership of participants in certain demographic group can be identified from different behavioral activities.
- **Demographic menagerie:** Existence of groups of users who perform differently at various authentication tasks has been shown in literature [188]. Demographic or physiological links to these groupings can be explored.

Table 2.16: Summary of Flight4 feature statistics. All values are in milliseconds.

Digraph	Desktop		Tablet		Phone	
	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$	$\mu(\text{avg})$	$\mu(\text{std})$
<i>d1</i>	380	268	335	230	336	173
<i>d2</i>	441	249	587	302	552	301
<i>d3</i>	495	247	656	299	644	285
<i>d4</i>	459	246	598	294	620	310
<i>d5</i>	436	239	595	313	577	292
<i>d6</i>	327	174	363	188	342	167
<i>d7</i>	343	166	340	135	324	122
<i>d8</i>	315	132	294	82	317	68
<i>d9</i>	369	132	337	89	339	82
<i>d10</i>	336	161	342	155	354	144
<i>d11</i>	320	123	404	150	410	122
<i>d12</i>	362	141	350	95	351	78
<i>d13</i>	296	106	282	79	295	64
<i>d14</i>	339	172	376	222	348	166
<i>d15</i>	340	117	319	72	338	67
<i>d16</i>	337	165	402	210	367	150
<i>d17</i>	300	107	303	83	307	68
<i>d18</i>	296	110	264	79	284	77

## 2.5 Comparison with other data sets

The datasets that are currently available for behavioral biometrics are collected with the focus on a single activity. We summarize and compare our dataset to other related datasets that are available in literature. We present the key points of comparison with sizeable and related datasets in Table 2.17.

A majority of keystroke datasets are focused on fixed text data with short strings like password, repeated many times by each participant [12, 87, 106]. Most keystroke datasets are collected on desktops [21, 87, 173, 182] and very few are on hand-held devices, such as [12], which has a only 42 participants of which 37 participants used tablets and only 5 used phones. Such variations limit the usability of datasets. For gait data, there are

Table 2.17: Comparison with other related datasets.

	Dataset	No. of users	Type of data	Type of activity	Device(s) used by participants	Highlights	
Keystroke	U@B_KD [173]	148	Latencies	Fixed & Free text	Desktop	Four different keyboards used	
	SBrook_KD [21]	196	Latencies	Free text	Desktop	Truthful Vs. Deceptive writing	
	Video_KD [58]	30	Latencies & Video	Fixed text	Desktop	Movement and Motor aspects	
	Pressure_KD [106]	100	Latencies & Pressure	Fixed text	Desktop	Pressure on desktop keyboard	
	Android_KD [12]	42	Latencies & Pressure	Fixed text	37 Tablet & 5 Phone users	Addition of pressure features for Android	
	Laser_2012 [88]	20	Latencies	Fixed & Free text	Desktop	Free vs Fixed text behavior	
	CMU_KD [87]	51	Latencies	Fixed text	Desktop	Large number of repetitions by participants	
	Clarkson_I [182]	39	Latencies & Video	Fixed & Free text	Desktop	Video of face and hand while typing	
	Clarkson_II[124]	103	Latencies	Fixed & Free text	Desktop	Natural and uncontrolled includes mouse and app data	
	Gait	Kinematics [63]	42	3d Motion Capture using Force Plates	Overground & Treadmill	Force Plates (placed on body)	Anthropometric data & Pelvis Kinematics
HuGaDB [42]		18	Accelerometer X 6 Gyroscope X 6 EMG X 2	12 Activities including Gait, Upstairs and Downstairs	Accelerometer, Gyroscope & EMG sensors (placed on body)	Body sensor network data	
CASIA-B [193]		124	Video	Walking	-	Effect of Viewpoint, Clothing & Carrying	
USF humanID [151]		122	Video	Walking	-	12 Experiments with changes in conditions.	
HAR [11]		30	Accelerometer & Gyroscope	6 Activities including Gait, Upstairs and Downstairs	Smartphone	Activity recognition using smartphone	
UniMiB SHAR [114]		30	Accelerometer	Gait & Fall	Smartphone	Fall detection	
OU-ISIR [132]		744	Accelerometer, & Gyroscope & Video	Gait & Slope	Inertial Sensor, & Smartphone	Large gait dataset with video & inertial sensors	
Swipe		ASU_Touch[98]	75	X, Y coordinates Pressure Area of touch Orientation of finger	Swipe & Touch Gestures	Smartphone	Re-authentication using swipe gestures
		Touchalytics[62]	41		Swipe to scroll through images	Smartphone	Proposed 30 touch features
		LTU_Touch[157]	190		Swipe through questions	Smartphone	Evaluation of verifiers for touch data
	FAST[59]	40	Browsing		Smartphone	Fingergestures Authentication System using Touchscreen (FAST)	
Our Dataset	Keystroke	Same 117 participants across the datasets	Latencies	Fixed and Free text	Desktop, Tablet & Phone	The same participants performing multiple, common day-to-day activities on multiple devices with real-life placement and usage of devices	
	Gait		Accelerometer X 3 Gyroscope X 3	Walking, Upstairs & Downstairs	Tablet in hand Phone in hand Phone in pocket		
	Swipe		X, Y coordinates Pressure, Area of touch Orientation of finger	Swipe through questions	Tablet Phone		

two datasets that provide sub-activities, similar to our dataset, such as walking, going up and down the stair case [11, 114]. But, both these datasets have only 30 participants. *HuGaDB* [42], provides a dataset with 12 different activities collected with a body-sensor-network having 6 accelerometers and gyroscopes each and 2 electromyography (EMG) sensors that are placed on the participants body. Though this data approximates the body movements of participants very closely, it would not be suitable for continuous authenti-

cation due to the unrealistic placement of sensors compared to day-to-day use of phones. In case of *CASIA B* and *USF HumanId* ([193], [151]), both having over 120 participants, a third party surveillance approach is more suitable as the datasets consist of video recordings of participants' gait which is not suitable for on-device continuous authentication.

The *OU-ISIR* [132] dataset is a large gait dataset consisting of more than 744 users from a wide age range of 2 to 74 years. The dataset also has synchronized video and inertial sensor (stand-alone or in smartphone) data for gait and stop-up and down activities.

As touch and swipe as a behavioral biometric is comparatively less explored and the number of datasets is far fewer. All touch and swipe datasets compared in Table 2.17 collected similar raw data (coordinates, pressure, area and orientation) while using different ways to make participants perform the swipes.

All the datasets discussed above provide data for single activity on single device. How a particular activity from a user varies from device to device, or existence of correlation between different activities on different devices cannot be explored with these datasets.

Heterogeneity Human-Activity Recognition dataset [169] consists of data from multiple devices and multiple movement-related activities (no keystrokes), but as data was collected from only nine users and the device carrying conditions (eight phones carried together in a pouch at the waist and two watches worn on each hand) limit the usability of the dataset for behavioral biometrics.

Therefore, our dataset stands unique by providing data from the same 117 participants performing; a) typing activity, both fixed and free text, on desktop, tablet and phone; b) gait activity, including walking, upstairs and downstairs, with phone in hand, tablet in hand and phone in pocket; and c) swiping activity on tablet and phone.



## 2.6 Lessons learnt from collection of the dataset

The process of collection, curation, pre-processing, storage and sharing of a dataset is indeed challenging. We dealt with various issues at each stage of this effort and share the key lessons learnt, for the benefit of research community, below:

- Overhead time for the entire process is nontrivial. Time involved in various legalities; preparation and approval of Institutional Review Board (IRB) documents; reaching out and procuring participants; and scheduling them is nontrivial and require great consideration and planning beforehand.
- Special attention must be given to avoid Personally Identifiable Information (PII) trickling into data. Especially when data collection aims to capture free text, participants might unknowingly divulge personal information such as names, phone numbers, email ids etc., as part of their answers to questions. This can be corrected either in the protocol designing phase (careful consideration to questions) or in the pre-processing phase.
- An on-site proctor to oversee each participant's data collection can ensure quality of data especially when data involves capturing key timestamps for activity separation (Section 2.2.3, Checkpoint files). In a few cases we have made manual corrections (by adding or subtracting seconds noted by proctor) to the timestamps where a participant logged them either too early or too late.
- When data collection involves logging of timestamps on more than one device it is important to make sure the clocks on all devices involved are synchronized to within a few milliseconds of each other. We carried out several test runs to ensure synchroniza-

tion of the timestamps on all devices, which was challenging as there were four devices (desktop, pocket-phone, hand-phone and tablet) to be used by every participant.

- Before sharing the data publicly, it is important to represent similar data from different devices in the same format for easier usability. For example; timestamps were logged in a string format (yyyy-mm-dd hr-min-sec.milliseconds) on the desktop and UNIX timestamp on all other devices; and key strokes were logged as characters or keys on the desktop but as ASCII codes on other devices. Therefore, it is better to standardize the data fields before sharing the dataset.
- Incomplete data can occur from unexpected application or sensor fault or when participants do not complete the entire process. For example, in our dataset, user 117 did not complete the tasks **h** to **m**, but other tasks were complete and are included in the final dataset. In rare cases, where there was too little information for a task or activity, it was better to remove the files for completeness of the shared data.
- In a previous data collection effort for keystrokes [156], we observed some participants tend to fill in low-quality or gibberish text in order to satisfy the minimum text-length criteria (if any) to finish the session earlier. Such occurrences reduce the quality of data and can be remedied either by clearly stating the dos and don'ts to the participants or by the on-site proctor observing and interrupting such behavior.

## 2.7 Conclusion and future work

Through this paper, we share and provide the details of our large behavioral biometrics dataset for typing, gait and swiping activities of the same user on desktop, tablet and phone (Section 2.2). The availability of the data on different devices for the same person makes our dataset unique; and with data from 117 participants, also one of the largest. With this dataset researchers can try to explore questions that were not possible with previously available datasets such as; *"Does the typing of an individual on desktop reveal their typing on a tablet or phone? and vice versa"* ; *"Can a person's demographics like age, height, etc., be predicted from the data of typing, gait or swiping activity on any of the devices?"*; to name a few. Each of the files in our dataset are described in detail with example snippets for easier visualization and understanding (Section 2.2.3).

We explore, describe, extract and analyze the most popular features for each activity in our dataset. All features are described briefly and also included with the dataset repository (Section 2.2.3). The demographics of the participants is shared and includes various physiological and background information with good spread for most groups (Section 2.2.4). The analysis of the features reveals interesting insights. We found participants took more time between keys on phones and tablets when compared to a desktop keyboard but the release event was much sooner on the phones and tablets implying that smaller amount on time is spent with the finger on the key (Section 2.3). As the general style of typing on tablets and phones is with just the two thumbs as opposed to several fingers on desktop, we posit that this occurrence maybe a result of lesser number of fingers being in contact

with the typing surface on hand-held devices thus increasing the keyhold time and reducing the inter-key latency on desktop.

This dataset helps address the scarcity in benchmark datasets for multi-device, multi-activity and multi-modality data from the same participants. Collection of a high-quality dataset that can be publicly shared for the benefit of the community, is indeed a tedious and demanding task. Throughout the process, lessons that we have shared in Section 2.6 are intended for future researchers who make similar endeavors to have an advantage. We also discuss several possible research directions (Section 2.4) that can be explored with the help of this dataset. As part of our future work we will be exploring these directions.

### **3. DISCRIMINATIVE POWER OF TYPING FEATURES ON DESKTOPS, TABLETS AND PHONES FOR USER IDENTIFICATION**

Research in Keystroke-Dynamics (KD) has customarily focused on temporal features without considering context to generate user templates that are used in authentication. Additionally, work on KD in hand-held devices like smart-phones and tablets have shown that these features alone do not perform satisfactorily for authentication. In this work, we analyze the discriminatory power of the most used conventional features found in literature, propose a set of context-sensitive or word-specific features and analyze the discriminatory power of proposed features using their classification results.

Typing is a common form of interaction, where a person provides input for these devices either on keyboards or touch screens, thus making research in Keystroke Dynamics (KD) popular. Research in KD has grown far and wide, Umphress and Williams [180], in their work, demonstrated that keystroke behavior on keyboards/typewriters was indeed a distinguishable trait among users while more recent research has shown that KD can also be used on other devices that involve typing, such as phones and tablets [47], [126]. A considerable amount of research has also explored the effects of the type of text used for KD, that is fixed text vs free text [4]. The problem of authenticating users by their typing behavior has also been addressed from multiple perspectives as far as the underlying algorithms are concerned. Fuzzy logic [94], Neural Networks [8], mini-batch bagging

[73], pairwise user coupling [118] techniques have been explored in an attempt to improve accuracies and complexity of KD systems. New approaches that use special hardware ([171], [186]) for KD systems are also a promising avenue for researchers. One such work from Sulong et al. [172] explored new features such as a combination of maximum pressure exerted on the keyboard and time latency between keystrokes and showed their proposed system was an effective biometric-based security system.

In recent years KD has been used in a myriad of applications, such as, continuous authentication [117], gender detection [177], age detection [138], fatigue detection [178], mood disturbance detection [198], and lie detection [115] to name a few. There have also been numerous attempts on side channel attacks on keystrokes based on the acoustic emanations that occur when a person types on physical keyboards. Asonov and Agrawal [16] trained neural networks to recognize the key pressed using the sounds emanated by their press. In their work, they used FFT on 2ms windows sound recordings and tested with recordings from varying distance from 1 meter to 15 meters. A similar work carried out by Zhu et al. [195] explores attacks using the acoustic emanations assuming that the different keys pressed are not contextually related. Using the Time Difference of Arrival (TDoA) method, they were able to recover about 72% of the keystrokes. With a modification of TDoA approach with mm-level audio ranging on a single phone, Liu et al. [104], were able to recover 94% keystrokes in their experiments. Although research in KD has been advancing rapidly, there have been very few attempts to understand the impact of context on the features that are used for KD and even the few attempts made were not exhaustive enough.

### 3.1 Key contributions on the chapter

The key contributions of our research work detailed in this article can be summarized as follows:

- **Analyze the discriminatory power of conventional keystroke features for user identification across 3 most common devices:** We present our findings, from analysis of conventional KD features, on the three most commonly used devices desktop, Laptop and phone. We find that conventional features do not separate and hence identify a user's keystroke data efficiently and are simplistic in disregarding the context of these features.

- **Propose, analyze and evaluate a new set of context-sensitive features across devices:**

We propose a set of context-sensitive or word-specific features, after analyzing the difference in the discriminative capacity of these features in contrast to the conventional features, we find that context sensitive features are better for user identification on all three devices. The results of user identification show competitive accuracies using proposed features.

- **Provide insights into efficacy of features for continuous authentication on different device categories:** We also provide mathematical justification of the performance improvements in user identification using proposed features.

KD on desktops, tablets and phones, which are the three most popular types of devices that people use everyday, are all analyzed and reported. This work will help gain a deeper

insight into what features might work better for continuous authentication on different device categories. Some initial applications are in the field of continuous authentication of user or as a second factor authentication in existing systems with typing as one of the interfaces. Scenarios like online examinations, competitions, remote work environments might benefit from the insights provided by our work. Situations where context is known can be handled much more accurately with the help of the proposed features described here. This article is also aimed at inspiring exploration of novel features that use additional information such as the language and contexts of features to their advantage.

### **3.2 Related Work**

Researchers of KD have explored and studied the effectiveness of various keystroke features for a long time now ([176], [139], [155]). In numerous research, available in already existing literature, it is clear that KD is a promising dimension for authentication and verification ([79], [119], [153], [82], [143], [43], [7], [160]). Initial work of Obaidat and Sadoun [130], on the analysis of features for verification of users based on keystroke dynamics is a continuation of three previously published papers by the authors. They demonstrated the advantages of using both, KeyHold value and flight time, as opposed to just one of them. They showed that there was a significant dip in misclassification errors when the features were used in combination. The study consisted of 225 samples collected each day, for eight days, from 15 users. Each sample was seven characters in length on average. The authors claimed 100% accuracy on this set of users using fuzzy ARTMAP, Radial Basis Function Network and learning vector Quantization. The main



drawbacks of this study were the limited number of users, the small text set of seven characters each and confining the data to only one device that is the desktop. Huang et al. [76] made one of the most recent efforts to analyze the effects of text filtering on keystroke biometrics. The authors have used the work of Gunetti and Picardi [71] as their baseline to present their case on the effect of text filtering. It is worth noting that this study concluded that nearly 23.3% of all free text keystrokes were gibberish. Gibberish was defined by the authors as text belonging to four main categories: Repetitive, Gaming ("a", "s", "d" and "w"), Distinct (too few distinct characters) and Lengthy (long strings with length more than 20). It is also shown that the density curves of many digraphs changed drastically after filtering. The authors' concluding remark is that filtering of gibberish has no effect on FAR but significantly improves FRR. Using two main filtering techniques, Regular Expressions and spell checkers, this study establishes that the context from which a feature is extracted plays an important role in the performance of the keystroke based system. Balagani et al. [19] analyzed the discriminability of heterogeneous and aggregate feature vectors with different combinations of keystroke features. The authors used ReliefF, correlation based feature selection, and consistency based feature selection to perform feature selection analysis. This work provided theoretical proof backed by empirical analysis to confirm that heterogeneous feature vectors were more discriminative than aggregate feature vectors.

Alsultan et al. [9] explored non-conventional keystroke features. As the authors call them, "Semi-Timing" and "Editing" features, and their advantages, were analyzed in this work. Semi-timing features defined were the Words Per Minute (WPM), Negative Up-Down (press second key, before releasing first) and Negative Up-Up (release second key, before

releasing first). The Editing features defined regarded the general tendencies and mannerisms of a user, like Error rate (number of backspaces), Caps-lock Usage and Shift Usage (resealed before/after letter). The use of Ant Colony Optimization (ACO) feature reduction, leading to 5 features that contributed the most (Negative Up-Down, Error Rate, Right-Shift-Before, Left-Shift-After, Left-Shift-Before) is a valuable insight. Decision trees and Support Vector Machines were used for a comparative study on classifier performance; Decision Trees had a slightly better performance, possibly due to the inbuilt feature reduction property of Decision Trees. The study presents competitive False Acceptance Rates (FAR) and False Reject Rates (FRR), 0.011 and 0.26 respectively, on a dataset of 30 users. Numerous researchers (Azevedo et al. [18], [52]) have also explored different feature selection techniques, distance measures and effects of different languages [72] on a KD system. Sun et al. [174], in their work, described of a group of secondary features such as shift and comma which had been previously overlooked as noise and also explored their effectiveness for user classification.

### **Related context-based work**

The effects of linguistic context on KD was explored by Goodkind et al. [67], in their work authors raised several important questions regarding the treatment of keystroke features, with respect to word boundaries and part-of-speech. Another research that comes close to our work was carried out by Sim and Janakiraman [163], in which the authors showed that features extracted from Di-Graphs and Tri-Graphs were not discriminative enough in free text. The authors rightly suggest that embedding of these features play a

role in their effectiveness. In both, authors fall short of proposing any word based features and also limit their studies to desktops only.

We show with empirical analysis that conventional features for KD are inadequate and can be greatly improved by factoring in knowledge of the language being typed. We propose a set of context based features and draw mathematical insights for their better performance. We perform our analysis on the data collected from three of the most common devices that the current populace interacts with: the desktop, the tablet and the phone. From our literature survey we also note that the search for optimal features for KD has not been exhausted, and the research community is actively pursuing analysis of non-conventional feature extraction. This is one of the incentives for us to propose our word-specific features.

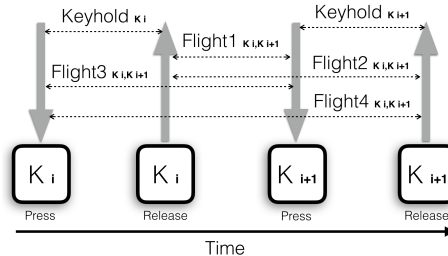


Fig. 3.1.: Features extracted from the temporal data of keys  $K_i$  and  $K_{i+1}$ .

### 3.3 Conventional keystroke features

We consider all the temporal features that are the building units of conventional features for KD (see Teh et al. [176]). All research on KD has the logging of keystrokes in common. When a user types, a log of each key pressed and released is stored along with the

timestamps of these events, from which a host of features are extracted. A Uni-graph is any single key being pressed and released. Similarly, a Di-graph is any two consecutive key being pressed and released. Uni-graphs and Di-graphs are the basic entities of KD and can be used to understand the temporal features. For example, if a user types "the", then "t", "h" and "e" are the Uni-graphs and "th" and "he" are the Di-graphs. For any Di-graph formed by keys  $K_i$  and  $K_{i+1}$  the following temporal features can be extracted:

- a.  $KeyHold_{K_i} : K_iRelease - K_iPress$
- b.  $KeyHold_{K_{i+1}} : K_{i+1}Release - K_{i+1}Press$
- c.  $Flight1_{K_iK_{i+1}} : K_{i+1}Press - K_iRelease$
- d.  $Flight2_{K_iK_{i+1}} : K_{i+1}Release - K_iRelease$
- e.  $Flight3_{K_iK_{i+1}} : K_{i+1}Press - K_iPress$
- f.  $Flight4_{K_iK_{i+1}} : K_{i+1}Release - K_iPress$

where  $K_iPress$  and  $K_iRelease$  correspond to the time when  $K_i$  was pressed and released respectively and so on. Features (a) and (b) are from Uni-graphs, (c) - (f) are from Di-graphs. We can also observe that the values of  $Flight1$  and  $Flight2$  could be negative as it is common for users to press and release a second key before the release of the first key. Figure 3.1 illustrates these temporal features which are integral to the discussions throughout this article.

Conventional KD research use features that are formed out of these basic temporal features. Each feature is generally sorted into separate groups based on characters that they

Table 3.1: Conventional features extracted from Uni-Graphs and Di-Graphs with their brief description.

Feature	Description
KeyHold	Time duration for which a key is held down.
Flight1	Time between release of first key and press of second key.
Flight2	Time between release of first key and release of second key.
Flight3	Time between press of first key and press of second key.
Flight4	Time between press of first key and release of second key.

Table 3.2: Conventional features extracted from an example string "this is that".  $U$ : Uni-Graph,  $D$ : Di-Graph.  $t_{1R}$  stands for release of key  $t_1$  (the subscript 1 stands for the first occurrence of "t") and  $t_{1P}$  stands for press of key  $t_1$  and so on.

$U$	KeyHold			
t	$(t_{1R}-t_{1P}), (t_{2R}-t_{2P}), (t_{3R}-t_{3P})$			
h	$(h_{1R}-h_{1P}), (h_{2R}-h_{2P})$			
i	$(i_{1R}-i_{1P}), (i_{2R}-i_{2P})$			
s	$(s_{1R}-s_{1P}), (s_{2R}-s_{2P})$			
-	$(-1R-1P), (-2R-2P)$			
a	$(a_{1R}-a_{1P})$			

$D$	Flight1	Flight2	Flight3	Flight4
th	$(h_{1P}-t_{1R}), (h_{2P}-t_{2R})$	$(h_{1R}-t_{1R}), (h_{2R}-t_{2R})$	$(h_{1P}-t_{1P}), (h_{2P}-t_{2P})$	$(h_{1R}-t_{1P}), (h_{2R}-t_{2P})$
hi	$(i_{1P}-h_{1R})$	$(i_{1R}-h_{1R})$	$(i_{1P}-h_{1P})$	$(i_{1R}-h_{1P})$
is	$(s_{1P}-i_{1R}), (s_{2P}-i_{2R})$	$(s_{1R}-i_{1R}), (s_{2R}-i_{2R})$	$(s_{1P}-i_{1P}), (s_{2P}-i_{2P})$	$(s_{1R}-i_{1P}), (s_{2R}-i_{2P})$
s_	$(-1P-s_{1R}), (-2P-s_{2R})$	$(-1R-s_{1R}), (-2R-s_{2R})$	$(-1P-s_{1P}), (-2P-s_{2P})$	$(-1R-s_{1P}), (-2R-s_{2P})$
.i	$(i_{2P}-1R)$	$(i_{2R}-1R)$	$(i_{2P}-1P)$	$(i_{2R}-1P)$
.t	$(t_{2P}-2R)$	$(t_{2R}-2R)$	$(t_{2P}-2P)$	$(t_{2R}-2P)$
ha	$(a_{1P}-h_{2R})$	$(a_{1R}-h_{2R})$	$(a_{1P}-h_{2P})$	$(a_{1R}-h_{2P})$
at	$(t_{3P}-a_{1R})$	$(t_{3R}-a_{1R})$	$(t_{3P}-a_{1P})$	$(t_{3R}-a_{1P})$

are derived from, for example, all KeyHold values for character "a" are grouped separately from those of character "b" and so on. Descriptive features are then extracted from these groups and stored as information for the user's template. The groups can be Uni-graphs, Di-graphs, Tri-graphs and so on. The features extracted and studied in this manner are referred to as "Conventional Features" in the following sections. Table 3.1 describes the conventional features that are widely used in literature. Conventional features,

as shown in table 3.1, are comprised of KeyHold, Flight1, Flight2, Flight3 and Flight4.

While KeyHold is extracted from a Uni-graph the rest are extracted from Di-graphs. A brief description for each feature is also provided in the table.

Table 3.2 demonstrates the extraction of these features with the help of an example string "this is that". The example string can be indexed as  $t_1 h_1 i_1 s_1 \_1 i_2 s_2 \_2 t_2 h_2 a_1 t_3$  (characters indexed per occurrence,  $\_$  represents 'space'). The conventional Uni-graph and Di-graph features extracted from the example string are simplistic in grouping together the values of Uni-graphs and Di-graphs disregarding the context of their occurrence. For example, KeyHold values of Uni-graph "t" are grouped. Conventional features do not distinguish where the values occur and most descriptive features derived from them will be aggregating values from entire pieces of text to be stored as templates for a user's typing behavior.

Table 3.3: Proposed context sensitive features and their brief description.

Feature	Description
WordHold	Time between the press of first key and the release of last key in the word.
AvgKeyHold	Average KeyHold values within a word.
AvgFlight1	Average Flight1 values within a word.
AvgFlight2	Average Flight2 values within a word.
AvgFlight3	Average Flight3 values within a word.
AvgFlight4	Average Flight4 values within a word.
StdKeyHold	Standard deviation of KeyHold in a word.
StdFlight1	Standard deviation of Flight1 in a word.
StdFlight2	Standard deviation of Flight2 in a word.
StdFlight3	Standard deviation of Flight3 in a word.
StdFlight4	Standard deviation of Flight4 in a word.

Table 3.4: Proposed features extracted from the same example string "this is that".  $t_{1R}$  stands for release of key  $t_1$  (the subscript 1 stands for the first occurrence of "t") and  $t_{1P}$  stands for press of key  $t_1$  and so on, Avg and Std stand for average and standard deviation respectively.

Feature	word: "this"	word: "that"
WordHold	$(s_{1R}-t_{1P})$	$(t_{3R}-t_{2P})$
AvgKeyHold	$\text{Avg}[(t_{1R}-t_{1P}), (h_{1R}-h_{1P}), (i_{1R}-i_{1P}), (s_{1R}-s_{1P})]$	$\text{Avg}[(t_{2R}-t_{2P}), (h_{2R}-h_{2P}), (a_{1R}-a_{1P}), (t_{3R}-t_{3P})]$
AvgFlight1	$\text{Avg}[(h_{1P}-t_{1R}), (i_{1P}-h_{1R}), (s_{1P}-i_{1R})]$	$\text{Avg}[(h_{2P}-t_{2R}), (a_{2P}-h_{2R}), (t_{1P}-a_{1R})]$
AvgFlight2	$\text{Avg}[(h_{1R}-t_{1R}), (i_{1R}-h_{1R}), (s_{1R}-i_{1R})]$	$\text{Avg}[(h_{2R}-t_{2R}), (a_{2R}-h_{2R}), (t_{1R}-a_{1R})]$
AvgFlight3	$\text{Avg}[(h_{1P}-t_{1P}), (i_{1P}-h_{1P}), (s_{1P}-i_{1P})]$	$\text{Avg}[(h_{2P}-t_{2P}), (a_{2P}-h_{2P}), (t_{1P}-a_{1P})]$
AvgFlight4	$\text{Avg}[(h_{1R}-t_{1P}), (i_{1R}-h_{1P}), (s_{1R}-i_{1P})]$	$\text{Avg}[(h_{2R}-t_{2P}), (a_{2R}-h_{2P}), (t_{1R}-a_{1P})]$
StdKeyHold	$\text{Std}[(t_{1R}-t_{1P}), (h_{1R}-h_{1P}), (i_{1R}-i_{1P}), (s_{1R}-s_{1P})]$	$\text{Std}[(t_{2R}-t_{2P}), (h_{2R}-h_{2P}), (a_{1R}-a_{1P}), (t_{3R}-t_{3P})]$
StdFlight1	$\text{Std}[(h_{1P}-t_{1R}), (i_{1P}-h_{1R}), (s_{1P}-i_{1R})]$	$\text{Std}[(h_{2P}-t_{2R}), (a_{2P}-h_{2R}), (t_{1P}-a_{1R})]$
StdFlight2	$\text{Std}[(h_{1R}-t_{1R}), (i_{1R}-h_{1R}), (s_{1R}-i_{1R})]$	$\text{Std}[(h_{2R}-t_{2R}), (a_{2R}-h_{2R}), (t_{1R}-a_{1R})]$
StdFlight3	$\text{Std}[(h_{1P}-t_{1P}), (i_{1P}-h_{1P}), (s_{1P}-i_{1P})]$	$\text{Std}[(h_{2P}-t_{2P}), (a_{2P}-h_{2P}), (t_{1P}-a_{1P})]$
StdFlight4	$\text{Std}[(h_{1R}-t_{1P}), (i_{1R}-h_{1P}), (s_{1R}-i_{1P})]$	$\text{Std}[(h_{2R}-t_{2P}), (a_{2R}-h_{2P}), (t_{1R}-a_{1P})]$

### 3.4 Proposed context sensitive features

In our approach, we focus on extracting descriptive features from words. For the purpose of simplicity, we limit our language to English, therefore all characters and words considered in this research are from English. We define words in the domain of KD as a set of consecutive keystrokes, preceded and succeeded by "space" or punctuation ("shift" is generally used for most punctuation), that form an English word. For example, if a user intends to type "the" but after typing "th" presses "a" by mistake and uses "backspace" to rectify this mistake, the sequence of keystrokes would be "t"+"h"+"a"+"backspace"+"e". Though the text on screen reads "the", the actual keystrokes performed were different;

hence we do not consider this the typing of a word. We use this concept whenever we refer to a "word" throughout this article.

Proposed features are shown in table 3.3, extracted for each occurrence of a word as opposed to Uni-graphs or Di-graphs and are comprised of "WordHold": the time taken to type an entire word (from first press to last release); "AvgKeyHold": the average of all KeyHold values in the word; "AvgFlight1", "AvgFlight2", "AvgFlight3" and "AvgFlight4": the average of the respective Flight values for Di-graphs in the word; "StdKeyHold": the standard deviation of all KeyHold values in the word; "StdFlight1", "StdFlight2", "StdFlight3" and "StdFlight4": the standard deviation of the respective Flight values for Di-graphs in the word. At this point in the discussion we have merely proposed these features and plan to investigate the efficiency of these proposed and conventional features as the discussion proceeds. Table 3.4 demonstrates the extraction of these features with the help of the same example string considered in table 3.2 : "this is that". The example string can be indexed as  $t_1 h_1 i_1 s_1 \_1 i_2 s_2 \_2 t_2 h_2 a_1 t_3$  (characters indexed per occurrence,  $\_$  represents 'space'). With our proposed set of features, we take advantage of the context by localizing the feature extraction to words in the given text. From each occurrence of a word, as shown in table 3.4, the proposed features do not group based on Uni-graphs and Di-graphs but rather based on words they appear in. The key takeaways from this section are that, conventional features do not distinguish where the values occur and most descriptive features derived from them will be aggregating values from entire pieces of text. Proposed features factor in the effect of context, by aggregating feature values only within the range of a word.



In following sections, we explain how conventional features are not ideal and overlook the rich information that lies in context of these features. We present thorough analysis of the discriminability of both conventional and proposed features and try and gain insights on why these features offer different levels of precision for user identification.




	 Desktop	 Tablet	 Phone
Tasks	Transcription (Fixed Text)		
	Browsing (Free Text) Q & A (Free Text)	Q & A (Free Text)	Q & A (Free Text)
Approx. Duration	45 mins.	25 mins.	25 mins.
Approx. Keystrokes per participant	12,500	9,000	10,000

Fig. 3.2.: Highlights of our Data Collection effort.

### 3.5 Details of data collection

Figure 3.2 summarizes the data collection process. After IRB approval from our university, the data collection exercise was carried out. We use the data from 20 users in this study. Emails were sent out to all students, faculty and staff to procure the participant population. All participants were proficient in English. Unlike most other studies, we did not restrict the type of device or text in our experiments. Each participant performed a set of common day to day activities on three different devices: a desktop, a tablet and a phone. Desktop I/O consisted of a standard QWERTY keyboard, optic mouse and 21” monitor. HTC-Nexus-9 tablets, Samsung-S6 and HTC-One phones were used in the process of data collection. The Samsung Galaxy S6 had a screen size of 5.1 inches with body

dimensions of 143.4 x 70.5 x 6.8 mm and weighing 138 g, whereas the HTC-One had a screen size of 5.0 inches with body dimensions of 146.4 x 70.6 x 9.4 mm and weighing 160 g. As the default android keyboard does not allow logging of keystrokes, we created and used an android qwerty keyboard on screen which was similar to the default android qwerty keyboard. The phones and tablets were locked in portrait orientation and users were allowed to type on them with any comfortable posture that they preferred. Although, there were no restrictions on the holding style of the phone and all participants chose to hold the phone with both hands while typing. For the typing activities both free text (spontaneous or unscripted typing) and fixed text (predetermined words or sentences to be typed as is) were used as a real life situation would comprise a mix of them.

For the desktop section of data collection, participants were asked to first type fixed text that consisted of two sentences, 20 times (Appendix A.5). This was followed by a brief session on browsing the Internet with tasks that approximated shopping behavior such as searching for the best prices and simultaneously making notes. The participants were then asked to type their free text answers to ten questions with varying cognitive loads [35] as shown in appendices A.2, A.3 and A.4. Keystroke events like press, release and their corresponding timestamps were recorded using windows keyboard hooks during the entire activity. For the tablet and phone section of the data collection, participants were asked to first type the same fixed text, 20 times. This was followed by typing free text answers to a set of ten questions with different cognitive loads. The questions used in each section were different. Keystroke and touch events with their corresponding timestamps were logged during the entire duration for both hand-held devices. The participants took about 45 minutes to complete the tasks on the desktop and 25 minutes each on the tablet

and phone. Each participant had approximately 12,500 keystrokes on the desktop, 9,000 keystrokes on the tablet and 10,000 keystrokes on the phone.

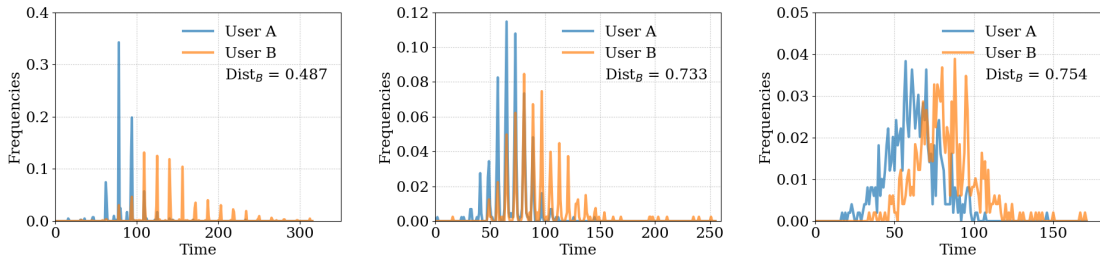
### 3.6 Feature discriminability analysis

To analyze the discriminative power of features, we first model an estimate of probability density function (PDF), given by the histograms of the values for each feature. We use fixed bin size of 1 ms for all histograms to match the clock resolution used to record keystroke events. To determine the discriminative potential of features we find the overlap between the PDFs using the Bhattacharyya distance as described by Sim and Janakiraman [163].

$$Dist_B(H1, H2) = \sum_{x=1}^{nbins} \sqrt{H1(x)H2(x)}. \quad (3.1)$$

Equation (3.1) defines the Bhattacharyya distance between two PDFs,  $H1(x)$  and  $H2(x)$ , in which the distance lies between 0 and 1. A  $Dist_B$  value of 0 implies no overlap in the PDFs, hence maximum discriminability and 1 implies complete overlap in the PDFs, hence minimum discriminability. As the PDFs we deal with are discretized, to implement equation 3.1 we simply multiply the probability of the corresponding bins, take the product's positive square root and sum it over all bins of the PDFs. Figure 3.3 shows an example of the computation of  $Dist_B$  values, using the PDFs of two random users for KeyHold values of the character "t" on different devices. Figure 3.3a shows the PDFs KeyHold of "t" for a desktop; the amount of overlap between the two is reflected by the  $Dist_B$  computed, which is 0.487. Similarly, figures 3.3b and 3.3c show the PDFs on a tablet

and phone which have  $Dist_B$  of 0.733 and 0.754 respectively. As explained in [163] a large  $Dist_B$  implies that the Bayes' error is large and a small  $Dist_B$  value could lead to small Bayes' error. We use  $Dist_B$  as a measure to analyze how well a feature separates the users from each other.



(a) KeyHold of "t" on desktop (b) KeyHold of "t" on tablet (c) KeyHold of "t" on phone

Fig. 3.3.: Example of  $Dist_B$  computation: Histograms representing the probability density functions of KeyHold values for the character 't', for Users A and B on a) desktop , b) tablet and c) phone along with their corresponding Bhattacharyya distance.

### 3.7 Analysis of conventional KD features

To analyze the discriminability of conventional KD features, we use the 12 most occurring Uni-graphs and 25 most occurring Di-graphs in our dataset. These Uni-graphs and Di-graphs are shown in tables 3.5 and 3.6 and include "space" and "backspace". For Uni-graphs and Di-graphs of each user we extract feature specific PDFs (one for each conventional KD feature in discussion). Once the PDFs are computed on all three devices, we calculate  $Dist_B$  between corresponding PDFs for each pair of users and use the mean, standard deviation and median to gain insight into the properties of these features.

### 3.7.1 KeyHold

**Description** : KeyHold is the time duration for which a key is held down for one instance of the key (one press to one release of the same key).

Table 3.5: The Inter-User  $Dist_B$  values for KeyHold distributions on all devices.

Device	Desktop			Tablet			Phone		
Uni-Graph	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
space	0.785	0.103	0.8	0.792	0.145	0.837	0.7	0.191	0.731
bspace	0.723	0.069	0.728	0.667	0.117	0.687	0.541	0.174	0.553
a	0.71	0.123	0.745	0.688	0.175	0.737	0.6	0.207	0.64
e	0.777	0.128	0.812	0.776	0.162	0.835	0.723	0.156	0.739
h	0.738	0.141	0.777	0.741	0.102	0.761	0.559	0.176	0.562
i	0.73	0.121	0.753	0.81	0.084	0.834	0.626	0.18	0.65
l	0.784	0.091	0.796	0.786	0.094	0.803	0.568	0.188	0.586
n	0.693	0.149	0.729	0.802	0.095	0.832	0.588	0.194	0.6
o	0.711	0.131	0.737	0.809	0.084	0.833	0.602	0.189	0.615
r	0.665	0.139	0.701	0.732	0.132	0.767	0.61	0.171	0.591
s	0.761	0.116	0.789	0.775	0.138	0.832	0.659	0.18	0.673
t	0.75	0.135	0.784	0.792	0.112	0.824	0.682	0.157	0.685

**Inference** : As values in the table 3.5 show the mean values of  $Dist_B$  for KeyHold are too high for all devices implying that this is not a very discriminative feature. Most mean  $Dist_B$  values for desktop lie around 0.7 to 0.78, which are hinting at very high overlap among the PDFs. The least and highest mean  $Dist_B$  were for "n" and "space" with 0.693 and 0.785 respectively. For tablet, the values  $Dist_B$  are very high, in the range of 0.7 to 0.8 for most characters. The least and highest mean  $Dist_B$  were "backspace" = 0.667 and "o" = 0.809 respectively. In the case of phone, we see a very negligible reduction in mean  $Dist_B$  values for phone, with most values between 0.58 to 0.72 which are still very high. The least and highest mean  $Dist_B$  were "backspace" = 0.54 and "e" = 0.72 respectively. By these values, we can infer that KeyHold, is not a discriminable feature on any of these devices.

### 3.7.2 Analysis of conventional feature - Flight1

**Description** : For a Di-graph, Flight1 is the time duration between release of the first key and press for the second key.

Table 3.6: The Inter-User  $Dist_B$  values Flight1 distributions on all devices.

Device	Desktop			Tablet			Phone		
Di-Graph	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
('space', 'a')	0.495	0.134	0.505	0.104	0.06	0.098	0.086	0.064	0.071
('space', 'i')	0.427	0.159	0.425	0.093	0.056	0.084	0.071	0.061	0.054
('space', 's')	0.526	0.12	0.54	0.135	0.064	0.128	0.081	0.056	0.065
('space', 't')	0.594	0.135	0.614	0.193	0.066	0.186	0.146	0.081	0.133
('bspace', 'bspace')	0.576	0.1	0.581	0.51	0.1	0.512	0.417	0.174	0.427
('e', 'space')	0.554	0.194	0.55	0.436	0.165	0.47	0.308	0.16	0.303
('e', 'n')	-			0.256	0.137	0.257	0.181	0.121	0.169
('e', 'r')	0.384	0.23	0.37	0.393	0.152	0.409	0.287	0.143	0.278
('e', 's')	0.555	0.171	0.587	0.429	0.154	0.445	0.331	0.137	0.333
('n', 'space')	0.405	0.211	0.429	0.361	0.136	0.357	0.217	0.139	0.192
('o', 'space')	0.379	0.241	0.345	0.275	0.13	0.264	0.188	0.133	0.168
('o', 'n')	0.412	0.219	0.464	0.353	0.158	0.385	0.269	0.14	0.253
('r', 'e')	0.434	0.27	0.465	0.472	0.145	0.492	0.378	0.154	0.371
('s', 'space')	0.499	0.206	0.501	0.363	0.142	0.378	0.228	0.141	0.227
('s', 'e')	0.531	0.234	0.572	0.59	0.124	0.62	0.39	0.159	0.38
('t', 'space')	0.481	0.223	0.465	0.325	0.113	0.34	0.248	0.127	0.232
('t', 'e')	-			0.267	0.132	0.253	0.237	0.135	0.223
('t', 'h')	0.511	0.242	0.523	0.462	0.158	0.5	0.344	0.152	0.357
('a', 'r')	0.399	0.231	0.407	0.261	0.124	0.247	-		
('t', 'o')	0.43	0.205	0.392	-			-		
('space', 'w')	0.351	0.159	0.365	-			-		
('h', 'e')	0.514	0.229	0.537	-			-		
('i', 'n')	0.346	0.218	0.361	-			-		
('l', 'e')	0.478	0.22	0.512	-			-		
('l', 'l')	0.671	0.216	0.762	-			-		

**Inference** : Table 3.6 shows the mean, standard deviation and median of the  $Dist_B$  values for flight1 on all three devices for the selected 25 most-common Di-graphs. Clearly, the mean values of  $Dist_B$  for Flight1 are considerably better than those for KeyHold on all devices. We observe that, most mean  $Dist_B$  values for desktop lie around 0.3 to 0.6, which is still not desirable for a feature meant to be discriminative. The least and highest mean  $Dist_B$  were for the digraphs (i,n)=0.346 and (l,l)=0.671 respectively. The mean

$Dist_B$  values for tablet are considerably low for a few Di-graphs, which is very desirable, but as these are a negligible minority (Space: a,i,s,t), their scope of extraction is largely reduced. The least and highest mean  $Dist_B$  were for the digraphs (Space,i)=0.093 and (s,e) =0.59 respectively. We observe the mean  $Dist_B$  values on phone to be similar to tablet with a small minority (Space: a,i,s,t) (e,n) of Di-graphs having low  $Dist_B$  values. (space,i) has the least mean value at 0.071 while has the highest at (Backspace,Backspace)=0.417. Overall Flight1 does not seem to be a very discriminative feature. As all Di-graphs have very high values of mean  $Dist_B$  on desktop and very few values in tablet and phone are at a desirable range this feature is not a good feature for to provide separation in user keystroke data.

### 3.7.3 Analysis of conventional feature - Flight2

**Description** : For a Di-graph, Flight2 is the time duration between release of the first key and release for the second key.

**Inferences** : Table 3.7 shows the mean  $Dist_B$  values for flight2 on all three devices for the selected 25 common Di-graphs. The mean values of  $Dist_B$  for Flight2 are considerably better than those for KeyHold on all devices, but are similar to Flight1. Most mean  $Dist_B$  values for desktop lie around 0.3 to 0.6. The least and highest mean  $Dist_B$  values being (Space,w)=0.382 and (space,t)=0.668 respectively. For the tablet, we observe that the mean  $Dist_B$  values for a few Di-graphs are considerably low, which is very desirable, but again, as these are a negligible minority (Space: a,i,s,t) their scope of extraction is largely reduced. The least mean  $Dist_B$  being (Space,i)=0.097 and highest being

Table 3.7: The Inter-User  $Dist_B$  values for Flight2 distributions on all devices.

Device	Desktop			Tablet			Phone		
	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
Di-Graph									
('space', 'a')	0.511	0.096	0.518	0.097	0.055	0.086	0.083	0.061	0.074
('space', 'i')	0.484	0.125	0.497	0.087	0.057	0.074	0.067	0.05	0.055
('space', 's')	0.556	0.109	0.563	0.119	0.057	0.107	0.078	0.051	0.071
('space', 't')	0.668	0.109	0.703	0.181	0.074	0.184	0.141	0.08	0.133
('backspace','backspace')	0.699	0.094	0.713	0.582	0.164	0.602	0.517	0.186	0.55
('e', 'space')	0.679	0.126	0.695	0.362	0.131	0.38	0.276	0.151	0.27
('e', 'n')	-			0.23	0.128	0.246	0.136	0.113	0.108
('e', 'r')	0.502	0.157	0.528	0.341	0.179	0.336	0.275	0.159	0.274
('e', 's')	0.586	0.144	0.624	0.37	0.159	0.383	0.275	0.139	0.271
('n', 'space')	0.466	0.192	0.49	0.308	0.125	0.317	0.177	0.135	0.162
('o', 'space')	0.475	0.229	0.502	0.235	0.109	0.247	0.176	0.133	0.154
('o', 'n')	0.453	0.156	0.465	0.305	0.146	0.332	0.22	0.144	0.214
('r', 'e')	0.563	0.195	0.621	0.455	0.145	0.472	0.356	0.175	0.342
('s', 'space')	0.59	0.168	0.611	0.293	0.11	0.298	0.211	0.141	0.202
('s', 'e')	0.599	0.185	0.64	0.538	0.153	0.572	0.371	0.176	0.357
('t', 'space')	0.556	0.161	0.576	0.264	0.095	0.271	0.223	0.111	0.204
('t', 'e')	-			0.264	0.109	0.254	0.208	0.126	0.187
('t', 'h')	0.615	0.18	0.657	0.439	0.144	0.453	0.294	0.154	0.286
('a', 'r')	0.49	0.179	0.522	0.224	0.114	0.22	-		
('t', 'o')	0.548	0.145	0.56	-			-		
('space', 'w')	0.382	0.143	0.395	-			-		
('h', 'e')	0.646	0.114	0.661	-			-		
('i', 'n')	0.411	0.187	0.436	-			-		
('l', 'e')	0.6	0.109	0.615	-			-		
('l', 'l')	0.639	0.169	0.692	-			-		

(backspace,backspace)=0.582. We observe phone to be Similar to tablet with a small number (Space: a,i,s,t) (e,n) of Di-graphs having low mean  $Dist_B$  values. The least and highest  $Dist_B$  values being (space,i)=0.067 and (Backspace,Backspace)= 0.517 respectively. Overall we infer that Flight2 is not a discriminative feature. As all Di-graphs have very high mean  $Dist_B$  values on the desktop and very few values in tablet and phone are desirable, this feature is not a good feature for separation of users based on keystroke data.



### 3.7.4 Analysis of conventional feature - Flight3

**Description** : For a Di-graph, Flight3 is the time duration between press of the first key and press of the second key.

Table 3.8: The Inter-User  $Dist_B$  values for Flight3 distributions on all devices.

Device	Desktop			Tablet			Phone		
Di-Graph	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
('space', 'a')	0.542	0.092	0.549	0.131	0.08	0.12	0.08	0.057	0.068
('space', 'i')	0.495	0.144	0.515	0.109	0.072	0.089	0.066	0.049	0.056
('space', 's')	0.58	0.105	0.594	0.15	0.078	0.146	0.078	0.051	0.069
('space', 't')	0.655	0.113	0.685	0.224	0.099	0.221	0.141	0.077	0.127
('bspace', 'bpace')	0.593	0.119	0.586	0.584	0.154	0.613	0.517	0.187	0.558
('e', 'space')	0.708	0.081	0.714	0.431	0.152	0.459	0.296	0.144	0.315
('e', 'n')	-			0.251	0.138	0.249	0.146	0.107	0.146
('e', 'r')	0.531	0.165	0.559	0.32	0.166	0.304	0.242	0.142	0.249
('e', 's')	0.643	0.089	0.654	0.387	0.175	0.396	0.275	0.12	0.272
('n', 'space')	0.508	0.167	0.547	0.35	0.132	0.342	0.154	0.113	0.126
('o', 'space')	0.524	0.211	0.578	0.305	0.146	0.291	0.173	0.121	0.161
('o', 'n')	0.493	0.179	0.532	0.3	0.147	0.306	0.216	0.112	0.221
('r', 'e')	0.652	0.17	0.699	0.457	0.141	0.463	0.352	0.149	0.364
('s', 'space')	0.639	0.113	0.656	0.377	0.133	0.39	0.209	0.134	0.197
('s', 'e')	0.639	0.151	0.659	0.537	0.176	0.563	0.382	0.16	0.382
('t', 'space')	0.604	0.14	0.631	0.321	0.121	0.314	0.232	0.113	0.219
('t', 'e')	-			0.259	0.124	0.243	0.178	0.12	0.158
('t', 'h')	0.654	0.151	0.691	0.49	0.122	0.495	0.297	0.141	0.307
('a', 'r')	0.525	0.153	0.558	0.241	0.125	0.221	-		
('t', 'o')	0.522	0.159	0.516	-			-		
('space', 'w')	0.397	0.151	0.414	-			-		
('h', 'e')	0.618	0.13	0.635	-			-		
('i', 'n')	0.423	0.191	0.469	-			-		
('l', 'e')	0.579	0.131	0.592	-			-		
('l', 'l')	0.633	0.194	0.7	-			-		

**Inference** : As shown in table 3.8, mean  $Dist_B$  values for flight3 on all three devices shown that Flight3 is considerably more discriminative than KeyHold on all devices, but is similar to Flight1 and Flight2. Most mean  $Dist_B$  values for desktop lie around 0.49 to 0.63, which is still high. The least and highest mean  $Dist_B$  values were for the Di-Graphs (Space,w)=0.397 and (e,space)=0.708 respectively. For tablet, the mean  $Dist_B$  values for a few Di-graphs are considerably low, similar to Flight1 and Flight2, which is

very desirable, but as these are a negligible minority (Space: a,i,s,t) their scope of extraction is largely reduced. The least and highest values for  $Dist_B$  were (Space,i) = 0.109 and (backspace,backspace) = 0.584 respectively. We observe phone to be Similar to tablet with a small minority (Space: a,i,s,t) (e,n) of Di-graphs having low  $Dist_B$  values. The lowest  $Dist_B$  value being (space,i) = 0.066 and highest being (Backspace,Backspace) = 0.517. We infer that, overall Flight3 is not a very discriminative feature. Most Di-Graphs have very high mean  $Dist_B$  values which implies that this will not be helpful in separation of the users based on their keystroke data.

### 3.7.5 Analysis of conventional feature - Flight4

**Description** : For a Di-graph, Flight4 is the time duration between press of the first key and release of the second key.

**Inference** : Table 3.9 presents the mean  $Dist_B$  values for flight4 on all three devices. These values are similar to those of all other Flight features that we have analyzed so far. The mean  $Dist_B$  values for Flight4 are better than those of KeyHold but not better than those of Flight1, Flight2 and Flight3. On the desktop, we find that most mean  $Dist_B$  values are in the range of 0.4 to 0.7 the least being (space,w) = 0.4 and highest being (e,space) = 0.74. On the tablet and on the phone, we see a few Di-Graphs having low mean  $Dist_B$  values but these are very few in number, (Space: a,i,s,t) on tablet and (Space: a,i,s,t) (e,n) on phone. The lowest and highest mean  $Dist_B$  values on tablet are for (Space,i) = 0.098 and (s,e) = 0.46 respectively while on phone they are (space,i) = 0.06 and (s,e) = 0.342 respectively. It is also interesting to observe that the digraphs

Table 3.9: The Inter-User  $Dist_B$  values for Flight4 distributions on all devices.

Device	Desktop			Tablet			Phone		
	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
Di-Graph									
('space', 'a')	0.514	0.089	0.522	0.106	0.064	0.099	0.083	0.056	0.077
('space', 'i')	0.498	0.12	0.506	0.098	0.059	0.088	0.06	0.043	0.053
('space', 's')	0.573	0.092	0.58	0.131	0.068	0.122	0.073	0.051	0.063
('space', 't')	0.671	0.096	0.69	0.199	0.09	0.185	0.137	0.076	0.132
('bspace', 'bpace')	0.579	0.1	0.593	0.421	0.138	0.432	0.34	0.178	0.345
('e', 'space')	0.74	0.071	0.75	0.336	0.118	0.355	0.267	0.141	0.268
('e', 'n')	-			0.218	0.12	0.214	0.122	0.098	0.105
('e', 'r')	0.576	0.122	0.589	0.285	0.168	0.288	0.215	0.139	0.204
('e', 's')	0.639	0.091	0.646	0.322	0.149	0.333	0.223	0.126	0.216
('n', 'space')	0.533	0.164	0.561	0.273	0.116	0.278	0.147	0.122	0.121
('o', 'space')	0.56	0.17	0.594	0.217	0.107	0.213	0.155	0.115	0.145
('o', 'n')	0.511	0.121	0.513	0.262	0.125	0.267	0.177	0.121	0.182
('r', 'e')	0.717	0.105	0.734	0.424	0.144	0.438	0.303	0.154	0.324
('s', 'space')	0.652	0.105	0.662	0.272	0.098	0.276	0.184	0.123	0.173
('s', 'e')	0.703	0.103	0.72	0.46	0.165	0.489	0.342	0.178	0.344
('t', 'space')	0.642	0.102	0.647	0.255	0.098	0.252	0.221	0.102	0.218
('t', 'e')	-			0.239	0.102	0.235	0.167	0.113	0.148
('t', 'h')	0.671	0.137	0.704	0.451	0.116	0.458	0.257	0.15	0.258
('a', 'r')	0.509	0.161	0.549	0.21	0.116	0.203	-		
('t', 'o')	0.572	0.142	0.585	-			-		
('space', 'w')	0.4	0.145	0.407	-			-		
('h', 'e')	0.65	0.108	0.679	-			-		
('i', 'n')	0.477	0.151	0.51	-			-		
('l', 'e')	0.627	0.107	0.647	-			-		
('l', 'l')	0.625	0.166	0.67	-			-		

with lowest and highest mean  $Dist_B$  values are for the same set of Di-Graphs on both the hand-held devices.

We illustrate the discriminability of the conventional features using CDFs of  $Dist_B$  shown in Figure 3.4. The CDFs help us compare the Bhattacharyya distances of each conventional feature among devices. Figure 3.4a shows that KeyHold value is not a discriminable feature on any of the devices. Though it has a slightly better curve on phone, it is still inefficient as a feature. From the Figure 3.4a we can estimate that less than 10% of the test samples have low (hence desirable) values of  $Dist_B$  of 0.2 or lesser. Around 60% of the test samples have  $Dist_B$  greater than 0.75 on desktop and tablet, and 0.5 on phone.

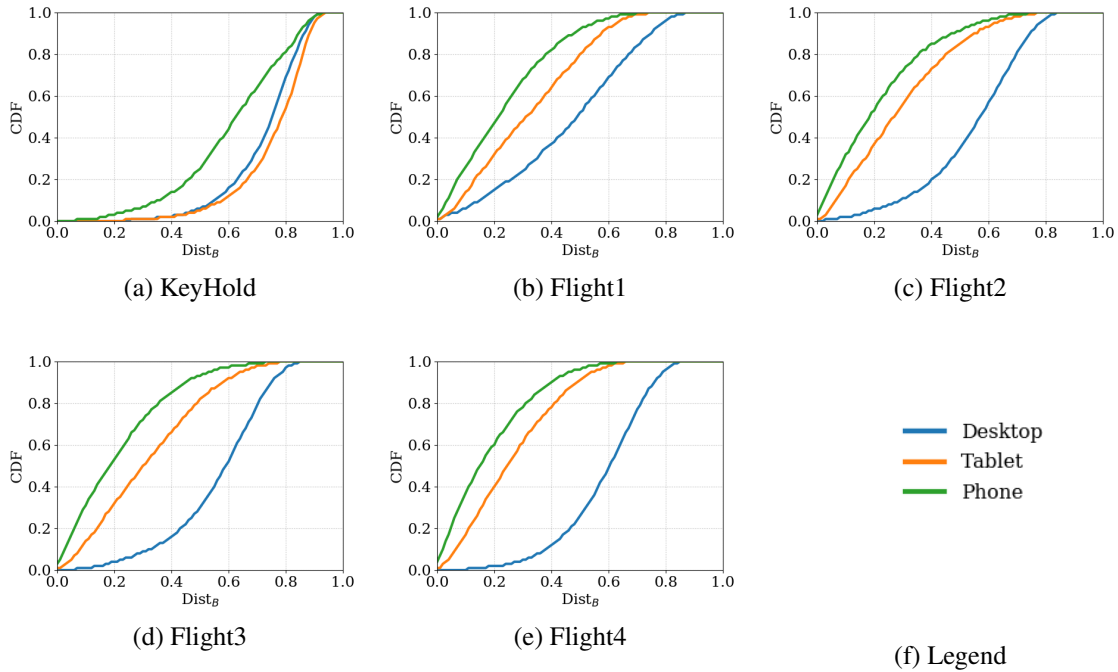


Fig. 3.4.: Comparing the Bhattacharyya distances of PDFs for all conventional features on desktop, tablet and phone.

These imply a very high overlap of the PDFs for this feature which makes it a weak feature.

For the features Flight1, Flight2, Flight3 and Flight4, figures 3.4b through 3.4e show that these features have similar distribution of  $Dist_B$  values with minor variations. A clear observation is that, though these features are clearly better than the KeyHold feature and that they have poor discriminability on the desktop data. Flight1 has some noticeable improvement with about 60% of them having  $Dist_B$  less than 0.5, while Flight2, Flight3 and Flight4 offer negligible improvements by having 40% with less than 0.5 for  $Dist_B$ .

For tablet and phone, there is improvement in discriminability on all the features, both have much better discriminability when compared to the desktop. We observe that the features seem to be slightly more discriminative on phone compared to the tablet. Flight1

(Figure 3.4b), we see about 60% of the test samples have  $Dist_B$  less than 0.4 on tablet and less than 0.3 on phone. In the case of Flight2 (Figure 3.4c) we see about 60% of the test samples have  $Dist_B$  less than 0.3 for tablet and less than 0.25 for phone. With Flight3 (figure 3.4d) we see about 60% of them have  $Dist_B$  less than 0.4 for tablet and less than 0.25 for phone, which is very similar to Flight2. Lastly, Flight4 (Figure 3.4e) shows 60% of  $Dist_B$  values are less than 0.3 for tablet and less than 0.2 for phone.

### **Inference - conventional features**

Though the Flight features have slightly better  $Dist_B$  values than KeyHold, none of these conventional features have high discriminative power which can be used to separate user keystroke data from each other to a high accuracy. We also observe that these conventional features aggregate all the values from the features without considering context, which might not be optimal, as variations in a feature value may occur due to the context of their appearance. We therefore propose a set of context sensitive features, which try and take advantage of known information, like the language being typed and the words of that language.

### **3.8 Analysis of proposed context sensitive features**

The proposed features are mentioned in table 3.3. These features are extracted for each occurrence of a word as opposed to each occurrence of the Di-graph, we extract "WordHold": the time taken to type an entire word (from first press to last release). "AvgKeyHold": the average of all KeyHold values in the word. "AvgFlight1", "AvgFlight2", "AvgFlight3",

”AvgFlight4”: the average of the respective flight values for Di-graphs in the word. ”Std-KeyHold”: the standard deviation of all KeyHold values in the word. ”StdFlight1”, ”StdFlight2”, ”StdFlight3”, ”StdFlight4”: the standard deviation of the respective flight values for Di-graphs in the word.

We analyze the discriminability of our proposed features by using the same methods that were used to analyze the conventional features. We selected 20 of the highest occurring words in our dataset, and generated the PDFs for each proposed feature, for every user on all three devices. We then computed the  $Dist_B$  values for the corresponding PDFs for every pair of users.

### 3.8.1 Analysis of proposed feature - WordHold

**Description** : WordHold is the time duration between the first key pressed to the last key released in a word. We only consider the sequence of keystrokes as forming a word if it is done without any deviations, such as backspaces or delete keys being pressed.

**Inference** : Table 3.10 presents the mean  $Dist_B$  values for WordHold on all three devices. The mean values of  $Dist_B$  for WordHold are considerably better than those of all the conventional features that were analyzed in the previous section on corresponding devices. Most mean  $Dist_B$  values for desktop are low and lie around 0.1 to 0.3, implying very less overlap among the PDFs. The highest mean  $Dist_B$  value being for ”that”=0.397. For tablet, the mean  $Dist_B$  values for all words were extremely low, which is very desirable, this implies that, the feature is good for differentiating between the users. All values are below 0.25 and majority of them are below 0.1. The highest mean  $Dist_B$  value

Table 3.10: The Inter-User  $Dist_B$  values for WordHold Distributions across all devices.

Device	Desktop			Tablet			Phone		
Word	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.391	0.175	0.4	0.099	0.092	0.081	0.069	0.073	0.046
carefully	0.072	0.064	0.074	-			0.026	0.04	0
data	0.305	0.135	0.323	0.012	0.027	0	0.034	0.05	0
different	0.153	0.077	0.143	0.01	0.024	0	-		
first	0.288	0.085	0.29	0.02	0.037	0	-		
have	0.221	0.182	0.239	0.028	0.057	0	0.024	0.042	0
lines	0.238	0.151	0.273	0.03	0.048	0	0.038	0.09	0
not	0.186	0.163	0.158	0.058	0.074	0.026	0.036	0.053	0
overlap	0.091	0.036	0.069	0.012	0.028	0	-		
phase	0.167	0.075	0.185	0.128	0.148	0.069	0.031	0.048	0
see	0.344	0.138	0.353	0.232	0.14	0.299	0.106	0.058	0.117
that	0.397	0.168	0.443	0.04	0.063	0	0.041	0.056	0
the	0.52	0.14	0.538	0.074	0.081	0.057	0.042	0.07	0
there	0.221	0.09	0.237	0.156	0.139	0.069	0.041	0.087	0
this	0.235	0.142	0.221	0.058	0.069	0.048	0.045	0.057	0.032
two	0.179	0.148	0.154	0.034	0.035	0.033	0.019	0.03	0
type	0.119	0.155	0.035	0.035	0.054	0	-		
will	0.193	0.153	0.192	0.035	0.046	0	0.013	0.027	0
with	0.173	0.146	0.139	0.025	0.041	0	0.022	0.035	0
words	0.133	0.104	0.121	0.069	0.107	0	0.005	0.018	0

was for "see" = 0.232. We observe phone to be similar to tablet with all words having extremely low  $Dist_B$  values. The least and highest mean  $Dist_B$  values are for "words"=0.005 and "see"= 0.106 respectively.

Overall the discriminability that this feature offers is very high. All three devices have very low values for mean  $Dist_B$  values. These values are especially low on hand-held devices and this appears to be a very good feature for user separation tasks like identification.

### 3.8.2 Analysis of proposed feature - AvgFlight1

**Description** : AvgFlight1 is the average of all Flight1 values occurring within the context of a word. Flight1 value of all Di-graphs in the word are summed and divided by the number of Di-graphs.

Table 3.11: The Inter-User  $Dist_B$  for AvgFlight1 Distributions across all devices.

Device	Desktop			Tablet			Phone		
Word	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.236	0.216	0.18	0.14	0.104	0.122	0.152	0.105	0.127
carefully	0.024	0.037	0	-			0.128	0.04	0.154
data	0.182	0.222	0.071	0.061	0.053	0.069	0.074	0.088	0.061
different	0.186	0.089	0.244	0.058	0.087	0	-		
first	0.181	0.1	0.148	0.039	0.061	0	-		
have	0.153	0.192	0.078	0.067	0.071	0.062	0.09	0.106	0.069
lines	0.122	0.153	0.036	0.077	0.064	0.069	0.15	0.105	0.143
not	0.094	0.131	0	0.094	0.087	0.075	0.043	0.054	0
overlap	0.046	0.036	0.069	0.037	0.059	0	-		
phase	0.186	0.146	0.215	0.069	0.062	0.069	0.205	0.094	0.196
see	0.277	0.216	0.267	0.205	0.067	0.215	0.102	0.104	0.067
that	0.165	0.152	0.105	0.088	0.091	0.065	0.085	0.094	0.067
the	0.403	0.231	0.442	0.123	0.102	0.104	0.098	0.116	0.06
there	0.101	0.205	0	0.056	0.086	0	0.102	0.151	0
this	0.177	0.148	0.176	0.113	0.085	0.101	0.126	0.107	0.114
two	0.131	0.105	0.101	0.045	0.052	0.033	0.018	0.04	0
type	0.192	0.215	0.17	0.08	0.061	0.101	-		
will	0.095	0.16	0.033	0.058	0.079	0	0.072	0.086	0.033
with	0.151	0.136	0.125	0.061	0.074	0.067	0.073	0.081	0.038
words	0.117	0.135	0.077	0.046	0.036	0.069	0.059	0.066	0.065

**Inference** : Table 3.11 presents the mean  $Dist_B$  values for AvgFlight1 on all three devices. The mean values of  $Dist_B$  for AvgFlight1 are better than those of all the conventional features and are comparable to those of WordHold on corresponding devices. Most mean  $Dist_B$  values for desktop are low and lie around 0.1 to 0.25, implying less overlap among the PDFs. This feature is more discriminable than the WordHold feature on desktop. For tablet again, the mean  $Dist_B$  values for all words were extremely low, which is very desirable. All values are below 0.25 and majority of them are below 0.1. We observe



phone to be similar to tablet with all words having extremely low  $Dist_B$  values. Overall the discriminability that this feature offers is also very high. All three devices have very low values for mean  $Dist_B$  values. These values are especially low on hand-held devices.

### 3.8.3 Analysis of proposed feature - AvgFlight2

**Description** : AvgFlight2 is the average of all Flight2 values occurring within the context of a word. Flight2 value of all Di-graphs in the word are summed and divided by the number of Di-graphs.

Table 3.12: The Inter-User  $Dist_b$  values for AvgFlight2 Distributions across all devices.

Device	Desktop			Tablet			Phone		
Word	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.415	0.21	0.449	0.159	0.117	0.155	0.126	0.105	0.109
carefully	0.083	0.014	0.074	-			0.164	0.071	0.186
data	0.41	0.161	0.417	0.068	0.072	0.067	0.068	0.115	0
different	0.123	0.045	0.124	0.075	0.119	0.03	-		
first	0.28	0.152	0.237	0.063	0.086	0	-		
have	0.305	0.21	0.322	0.047	0.076	0	0.103	0.141	0
lines	0.247	0.201	0.273	0.114	0.107	0.069	0.043	0.072	0
not	0.173	0.172	0.148	0.083	0.091	0.066	0.044	0.068	0
overlap	0.124	0.046	0.138	0.045	0.077	0	-		
phase	0.281	0.09	0.268	0.231	0.144	0.138	0.03	0.046	0
see	0.34	0.184	0.341	0.261	0.128	0.285	0.119	0.105	0.114
that	0.401	0.235	0.414	0	0	0	0.069	0.093	0.031
the	0.562	0.157	0.595	0.116	0.102	0.104	0.089	0.116	0.046
there	0.308	0.213	0.27	0.23	0.12	0.167	0.099	0.144	0
this	0.261	0.168	0.238	0.128	0.096	0.118	0.109	0.107	0.085
two	0.202	0.183	0.215	0.032	0.05	0	0.021	0.033	0
type	0.182	0.152	0.118	0.076	0.066	0.082	-		
will	0.181	0.177	0.169	0.063	0.096	0	0.03	0.05	0
with	0.184	0.158	0.151	0.063	0.057	0.071	0.08	0.092	0.035
words	0.186	0.094	0.191	0.145	0.168	0.077	0.038	0.052	0

**Inference** : Table 3.12 presents the mean  $Dist_B$  values for AvgFlight2 on all three devices. The mean values of  $Dist_B$  for AvgFlight2 are better than those of all the conventional features and are comparable to those of WordHold and the AvgFlight1 on corre-

sponding devices. Most mean  $Dist_B$  values for desktop are low and lie around 0.1 to 0.4. The highest mean  $Dist_B$  value was for "the" = 0.562 which seems to be a rare case as most values are lesser than 0.3. Again, in the case of hand-held devices we observe very low mean  $Dist_B$  values. For tablets the mean  $Dist_B$  values for all words were extremely low, which is very desirable, this implies that, the feature is good for differentiating between the users. All values being below 0.27 and majority of them lying below 0.1. It is intriguing that the word "that" had mean  $Dist_B$  value of 0, StD = 0 and median = 0, meaning no overlaps between any two user PDFs, theoretically providing 100% separation. We observe phone to be similar to tablet with all words having extremely low  $Dist_B$  values. Even the highest value is for the word "carefully" = 0.164, which is low. We infer that the Discriminability offered by this feature is high. All three devices have very low values for mean  $Dist_B$ , especially extremely low values on hand-held devices, it appears to be a very good feature for authentication/verification purposes.

### 3.8.4 Analysis of proposed feature - AvgFlight3

**Description** : AvgFlight3 is the average of all Flight3 values occurring within the context of a word. Flight3 value of all Di-graphs in the word are summed and divided by the number of Di-graphs.

**Inference** : Table 3.13 presents the mean  $Dist_B$  values for AvgFlight3 on all three devices. The mean values of  $Dist_B$  for AvgFlight3 are better than those of all the conventional features and are comparable to those of WordHold, AvgFlight1 and AvgFlight2 on corresponding devices. Most mean  $Dist_B$  values for desktop are low and lie around 0.1

Table 3.13: The Inter-User  $Dist_B$  values for AvgFlight3 Distributions across all devices.

Device	Desktop			Tablet			Phone		
	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.407	0.201	0.432	0.153	0.12	0.132	0.121	0.098	0.113
carefully	0.057	0.089	0	-			0.128	0.04	0.154
data	0.476	0.127	0.483	0.054	0.061	0.035	0.051	0.089	0
different	0.186	0.067	0.214	0.053	0.098	0	-		
first	0.232	0.166	0.194	0.04	0.075	0	-		
have	0.283	0.225	0.291	0.081	0.092	0.069	0.171	0.164	0.138
lines	0.299	0.193	0.311	0.061	0.06	0.069	0.07	0.099	0.037
not	0.174	0.189	0.089	0.109	0.105	0.081	0.049	0.064	0
overlap	0.132	0.029	0.138	0.035	0.037	0.033	-		
phase	0.219	0.112	0.225	0.171	0.169	0.138	0.132	0.055	0.126
see	0.453	0.152	0.458	0.346	0.105	0.302	0.078	0.116	0.033
that	0.304	0.196	0.371	0.097	0.104	0.065	0.075	0.093	0.033
the	0.504	0.173	0.54	0.134	0.115	0.116	0.088	0.109	0.051
there	0.281	0.145	0.324	0.253	0.111	0.237	0.089	0.15	0
this	0.294	0.189	0.266	0.131	0.113	0.11	0.106	0.107	0.076
two	0.249	0.199	0.225	0.011	0.026	0	0.032	0.043	0
type	0.159	0.171	0.131	0.092	0.065	0.074	-		
will	0.264	0.22	0.181	0.054	0.073	0	0.023	0.047	0
with	0.192	0.169	0.173	0.056	0.064	0.061	0.123	0.15	0.077
words	0.173	0.146	0.139	0.112	0.123	0.069	0.028	0.043	0

to 0.4. The highest mean  $Dist_B$  value was for the word "the" = 0.504, which is among the very few words with  $Dist_B$  greater than 0.4 for this feature. For tablets the mean  $Dist_B$  values for all words were extremely low. All values lie below 0.26 except for "see" and majority of them lie below 0.1. Again, we observe phone to be similar to tablet with all words having extremely low  $Dist_B$  values of less than 0.2.

Overall this feature offers high discriminability. All three devices have very low values for mean  $Dist_B$ . Once again, the values on hand-held devices are much lower than those of the desktop.

### 3.8.5 Analysis of proposed feature - AvgFlight4

**Description** : AvgFlight4 is the average of all Flight4 values occurring within the context of a word. Flight3 value of all Di-graphs in the word are summed and divided by the number of Di-graphs.

Table 3.14: The Inter-User mean  $Dist_B$  values for AvgFlight4 Distributions across all devices.

Device	Desktop			Tablet			Phone		
Word	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.354	0.169	0.367	0.127	0.099	0.106	0.11	0.104	0.09
carefully	0.123	0.077	0.074	-			0.166	0.036	0.154
data	0.18	0.186	0.128	0.037	0.059	0	0.064	0.09	0
different	0.024	0.037	0	0.111	0.125	0.065	-		
first	0.215	0.124	0.195	0.05	0.057	0	-		
have	0.228	0.133	0.182	0.047	0.068	0	0.057	0.076	0
lines	0.31	0.202	0.272	0.085	0.078	0.069	0.047	0.072	0
not	0.213	0.161	0.203	0.076	0.075	0.068	0.045	0.067	0
overlap	0.148	0.136	0.138	0.075	0.111	0	-		
phase	0.162	0.135	0.146	0.069	0.107	0	0	0	0
see	0.289	0.157	0.297	0.19	0.119	0.178	0.073	0.073	0.078
that	0.31	0.185	0.267	0.113	0.073	0.067	0.037	0.039	0.031
the	0.506	0.139	0.512	0.099	0.091	0.072	0.069	0.099	0
there	0.287	0.166	0.244	0.023	0.036	0	0.1	0.11	0.061
this	0.322	0.117	0.331	0.098	0.076	0.098	0.089	0.091	0.061
two	0.191	0.136	0.194	0.057	0.087	0	0.032	0.034	0.03
type	0.269	0.176	0.281	0.069	0.085	0	-		
will	0.293	0.112	0.281	0.053	0.059	0.033	0.055	0.08	0
with	0.212	0.132	0.202	0.073	0.061	0.074	0.097	0.093	0.076
words	0.142	0.076	0.113	0.095	0.034	0.077	0.025	0.036	0

**Inference** : Table 3.14 presents the mean  $Dist_B$  values for AvgFlight4 on all three devices. The mean values of  $Dist_B$  for AvgFlight4 are better than those of all the conventional features and are comparable to those of WordHold, AvgFlight1, AvgFlight2, AvgFlight3 on corresponding devices. For desktop most mean  $Dist_B$  values are low and lie around 0.1 to 0.4. For tablet and phone, the mean  $Dist_B$  values for all words were extremely low, which is very desirable. For tablet, all values lie below 0.2 and majority of them lie below 0.1. For phone all words have extremely low  $Dist_B$  values of less than

0.17. We again came across a peculiar case on phone, where for the word phase, theoretically mean of 0, StD = 0 and median = 0 implies no overlap between PDFs of any two users.

We infer that the feature, AvgFlight4, has high discriminability. All devices have very low values for mean  $Dist_B$  and extremely low values on hand-held devices.

### 3.8.6 Analysis of proposed feature - AvgKeyHold

**Description** : AvgKeyHold is the average of all KeyHold values occurring within the context of a word. KeyHold value of all Uni-graphs in the word are summed and divided by the number of Uni-graphs.

Table 3.15: The Inter-User  $Dist_B$  values for AvgKeyHold Distributions across all devices.

Device	Desktop			Tablet			Phone		
Word	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.281	0.247	0.233	0.31	0.17	0.316	0.25	0.182	0.247
carefully	0	0	0	-			0.382	0.209	0.371
data	0.143	0.159	0.128	0.145	0.155	0.096	0.259	0.153	0.242
different	0.234	0.179	0.172	0.181	0.21	0.101	-		
first	0.252	0.177	0.253	0.264	0.189	0.253	-		
have	0.231	0.192	0.176	0.182	0.17	0.139	0.198	0.146	0.211
lines	0.203	0.217	0.139	0.233	0.11	0.265	0.164	0.172	0.074
not	0.238	0.172	0.214	0.291	0.178	0.296	0.138	0.153	0.072
overlap	0.208	0.116	0.228	0.136	0.116	0.151	-		
phase	0.224	0.197	0.283	0.209	0.167	0.264	0.179	0.23	0.063
see	0.3	0.223	0.296	0.255	0.132	0.273	0.207	0.081	0.209
that	0.218	0.272	0.074	0.167	0.079	0.207	0.212	0.091	0.229
the	0.44	0.243	0.439	0.244	0.161	0.251	0.242	0.179	0.242
there	0.118	0.192	0.044	0.075	0.069	0.069	0.359	0.169	0.409
this	0.29	0.195	0.282	0.29	0.159	0.292	0.256	0.182	0.262
two	0.264	0.308	0.077	0.122	0.137	0.067	0.198	0.083	0.217
type	0.159	0.15	0.135	0.166	0.084	0.162	-		
will	0.248	0.238	0.211	0.161	0.111	0.143	0.268	0.137	0.27
with	0.277	0.218	0.259	0.165	0.172	0.105	0.272	0.095	0.242
words	0.146	0.134	0.126	0.026	0.04	0	0.228	0.186	0.2

**Inference** : Table 3.15 presents the mean  $Dist_B$  values for AvgKeyHold. The mean values of  $Dist_B$  for AvgKeyHold are better than those of all the conventional features but are inferior to those of WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4 on corresponding devices, except for desktop where they are similar to them. Most mean  $Dist_B$  values for desktop are low and lie around 0.1 to 0.3, implying less overlap among the PDFs. The word "carefully" has a mean, std and median  $Dist_B$  value of 0, therefore theoretically this feature should provide 100% separation among users. For tablet the mean  $Dist_B$  values are low but are slightly higher than their corresponding values in WordHold, AvgFlight1 through AvgFlight4. All values lie below 0.32 and majority of them are below 0.2. The least and the highest mean  $Dist_B$  value are for "there"=0.075 and "are" = 0.31 respectively. With phone we make similar observations as that of tablet  $Dist_B$  values and all values are less than 0.39. The least and highest values were for the words "not"=0.138 and "carefully"= 0.382 respectively. Overall, AvgKeyHold is not as discriminable as WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4 and does not provide much improvement from the conventional features.

### 3.8.7 Analysis of proposed feature - StdFlight1

**Description** : StdFlight1 is the standard deviation of all Flight1 values occurring within the context of a word.

**Inference** : Table 3.16 presents the mean  $Dist_B$  values for StdFlight1 on all three devices. Similar to the AvgKeyHold mean values of  $Dist_B$  for Std-Word-Flight1 are better than those of all the conventional features but are inferior to those of WordHold, AvgFlight1,

Table 3.16: The Inter-User  $Dist_B$  values for StdFlight1 Distributions across all devices.

Device	Desktop			Tablet			Phone		
Word	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.511	0.217	0.531	0.3	0.127	0.297	0.205	0.112	0.213
carefully	0.049	0.038	0.074	-			0.179	0.04	0.154
data	0.195	0.187	0.101	0.095	0.059	0.091	0.126	0.136	0.077
different	0.143	0	0.143	0.134	0.114	0.154	-		
first	0.171	0.121	0.164	0.094	0.103	0.074	-		
have	0.288	0.187	0.312	0.115	0.108	0.072	0.129	0.088	0.113
lines	0.112	0.12	0.073	0.13	0.103	0.139	0.197	0.074	0.199
not	0.405	0.289	0.475	0.229	0.137	0.215	0.138	0.116	0.108
overlap	0.023	0.036	0	0.024	0.035	0	-		
phase	0.059	0.065	0.068	0.218	0.073	0.207	0.232	0.175	0.148
see	0.448	0.203	0.499	0.465	0.154	0.476	0.226	0.089	0.199
that	0.268	0.123	0.257	0.268	0.124	0.194	0.147	0.123	0.146
the	0.585	0.17	0.615	0.233	0.143	0.22	0.227	0.127	0.224
there	0.168	0.161	0.148	0.069	0.062	0.069	0.172	0.113	0.198
this	0.332	0.142	0.367	0.186	0.109	0.177	0.198	0.126	0.184
two	0.349	0.175	0.414	0.183	0.092	0.201	0.116	0.092	0.088
type	0.177	0.101	0.175	0.075	0.067	0.07	-		
will	0.146	0.118	0.152	0.065	0.07	0.072	0.179	0.138	0.154
with	0.158	0.127	0.137	0.095	0.08	0.074	0.118	0.145	0.087
words	0.052	0.057	0.036	0.023	0.036	0	0.082	0.076	0.072

AvgFlight2, AvgFlight3 and AvgFlight4 on corresponding devices, except for desktop where they are similar to them. Most mean  $Dist_B$  values for desktop lie around 0.1 to 0.6, implying considerable overlap among the PDFs. For tablet the mean  $Dist_B$  values are low but are slightly higher than their corresponding values in WordHold, AvgFlight1 through AvgFlight4. All values are below 0.30 except for "see", and majority of them lie below 0.25. With phone we make similar observations as that of tablet  $Dist_B$  values, all values are less than 0.25. We can infer that StdFlight1 is not as discriminable as WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4.

### 3.8.8 Analysis of proposed feature - StdFlight2

**Description** : StdFlight2 is the standard deviation of all Flight2 values occurring within the context of a word.

Table 3.17: The Inter-User  $Dist_B$  values for StdFlight2 Distributions across all devices.

Device	Desktop			Tablet			Phone		
Word	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.454	0.187	0.471	0.258	0.11	0.26	0.23	0.128	0.218
carefully	0.049	0.077	0	-			0.051	0.04	0.077
data	0.195	0.163	0.161	0.102	0.172	0	0.145	0.134	0.154
different	0.095	0.037	0.071	0.155	0.094	0.201	-		
first	0.221	0.127	0.181	0.09	0.105	0.064	-		
have	0.184	0.141	0.181	0.13	0.081	0.129	0.117	0.102	0.072
lines	0.115	0.123	0.088	0.093	0.079	0.077	0.133	0.096	0.129
not	0.362	0.249	0.358	0.189	0.121	0.171	0.125	0.101	0.113
overlap	0.045	0.035	0.067	0.067	0.06	0.07	-		
phase	0.049	0.056	0.033	0.158	0.074	0.167	0.223	0.082	0.222
see	0.475	0.218	0.507	0.242	0.084	0.22	0.115	0.089	0.105
that	0.288	0.106	0.253	0.118	0.085	0.065	0.15	0.119	0.161
the	0.608	0.14	0.624	0.23	0.125	0.228	0.215	0.14	0.195
there	0.213	0.081	0.231	0.164	0.082	0.139	0.199	0.121	0.189
this	0.297	0.113	0.28	0.179	0.098	0.176	0.186	0.115	0.184
two	0.33	0.155	0.327	0.074	0.051	0.069	0.084	0.124	0
type	0.144	0.107	0.139	0.063	0.057	0.07	-		
will	0.215	0.108	0.23	0.085	0.08	0.069	0.159	0.17	0.072
with	0.202	0.113	0.196	0.115	0.091	0.134	0.125	0.13	0.086
words	0.1	0.091	0.088	0.118	0.032	0.139	0.117	0.108	0.077

**Inference** : Table 3.17 presents the mean  $Dist_B$  values for StdFlight2 on all three devices. The mean values of  $Dist_B$  for Std-Word-Flight2 are better than those of all the conventional features but are inferior to those of WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4 on corresponding devices, but for desktop they are similar to them. Most mean  $Dist_B$  values for desktop lie around 0.1 to 0.48, implying considerable overlap among the PDFs. For tablet the mean  $Dist_B$  values are low but are slightly higher than their corresponding values in WordHold, AvgFlight1 through AvgFlight4. All values are below 0.26, and majority of them lying below 0.2. With phone we make similar obser-



vations as that of tablet  $Dist_B$  values all of which are less than 0.24. Discriminability is not as high as WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4 using this feature. This feature also does not provide an improvement over conventional features.

### 3.8.9 Analysis of proposed feature - StdFlight3

**Description** : StdFlight3 is the standard deviation of all Flight3 values occurring within the context of a word.

Table 3.18: The Inter-User  $Dist_B$  values for StdFlight3 Distributions across all devices.

Device	Desktop			Tablet			Phone		
	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.59	0.135	0.577	0.287	0.115	0.287	0.194	0.101	0.193
carefully	0.074	0.066	0.074	-			0.149	0.121	0.186
data	0.282	0.141	0.295	0.068	0.118	0	0.108	0.096	0.109
different	0.071	0.064	0.071	0.185	0.087	0.164	-		
first	0.221	0.055	0.216	0.104	0.078	0.081	-		
have	0.269	0.168	0.226	0.149	0.071	0.153	0.111	0.101	0.082
lines	0.236	0.114	0.206	0.116	0.097	0.098	0.179	0.04	0.175
not	0.373	0.174	0.356	0.213	0.121	0.214	0.109	0.089	0.109
overlap	0.023	0.036	0	0.087	0.095	0.067	-		
phase	0.104	0.074	0.109	0.093	0.035	0.071	0.133	0.103	0.189
see	0.544	0.104	0.554	0.322	0.095	0.297	0.157	0.105	0.155
that	0.277	0.11	0.253	0.166	0.079	0.2	0.145	0.123	0.139
the	0.613	0.138	0.633	0.285	0.123	0.283	0.236	0.118	0.218
there	0.166	0.063	0.166	0.141	0.057	0.167	0.159	0.115	0.141
this	0.371	0.123	0.384	0.191	0.11	0.182	0.22	0.128	0.217
two	0.323	0.156	0.322	0.186	0.062	0.161	0.086	0.099	0.073
type	0.08	0.048	0.07	0.127	0.052	0.121	-		
will	0.185	0.135	0.152	0.061	0.158	0	0.23	0.173	0.182
with	0.198	0.115	0.173	0.136	0.123	0.092	0.104	0.094	0.077
words	0.082	0.071	0.074	0.195	0.087	0.139	0.078	0.095	0.065

**Inference** : Table 3.18 presents the mean  $Dist_B$  values for StdFlight3 on all three devices. The mean values of  $Dist_B$  for Std-Word-Flight3 are better than those of all the conventional features but are inferior to those of WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4 on corresponding devices, but again for desktop they are sim-

ilar. Most mean  $Dist_B$  values for desktop lie around 0.1 to 0.4, implying considerable overlap among the PDFs. For tablet the mean  $Dist_B$  values are low but are slightly higher than their corresponding values in WordHold, AvgFlight1 through AvgFlight4. All values lie below 0.35, and majority of them are below 0.3, The least and the highest mean  $Dist_B$  values were for "will"=0.061 and "see" = 0.322 respectively. The values for mean  $Dist_B$  are also low for phone. All values lie below 0.24 The least mean  $Dist_B$  value was for "words"=0.078 and highest for "the"= 0.236. Overall discriminability is not as high as WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4 using this feature. Does not provide desirable improvement from the conventional features.

### 3.8.10 Analysis of proposed feature - StdFlight4

**Description** : StdFlight4 is the standard deviation of all Flight4 values occurring within the context of a word.

**Inference** : Table 3.19 presents the mean  $Dist_B$  values for StdFlight4. The mean values of  $Dist_B$  for Std-Word-Flight4 are also better than those of all the conventional features but are inferior to those of WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4 on corresponding devices, but again for desktop they are similar. Most mean  $Dist_B$  values for desktop lie around 0.1 to 0.5, implying considerable overlap among the PDFs. The least and the highest mean  $Dist_B$  values are for "carefully"=0.025 and "the"=0.67 respectively. For tablet the mean  $Dist_B$  values are low, but are slightly higher than their corresponding values in WordHold, AvgFlight1 through AvgFlight4. All values being below 0.35, and majority of them lying below 0.3, least value for "data"=0.05

Table 3.19: The Inter-User  $Dist_B$  values for StdFlight4 Distributions across all devices.

Device	Desktop			Tablet			Phone		
Word	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.594	0.13	0.623	0.285	0.131	0.269	0.218	0.113	0.196
carefully	0.025	0.038	0	-			0.205	0.04	0.231
data	0.202	0.099	0.202	0.05	0.086	0	0.139	0.132	0.101
different	0.129	0.046	0.143	0.208	0.149	0.181	-		
first	0.179	0.095	0.151	0.087	0.074	0.077	-		
have	0.223	0.15	0.191	0.145	0.078	0.143	0.136	0.105	0.148
lines	0.111	0.086	0.103	0.115	0.104	0.098	0.153	0.128	0.111
not	0.437	0.143	0.42	0.237	0.143	0.227	0.123	0.102	0.099
overlap	0.033	0.05	0	0.091	0.09	0.068	-		
phase	0.066	0.071	0.069	0.103	0.053	0.069	0.146	0.113	0.215
see	0.573	0.102	0.569	0.292	0.129	0.274	0.139	0.09	0.147
that	0.364	0.137	0.316	0.15	0.122	0.182	0.131	0.072	0.14
the	0.67	0.131	0.685	0.236	0.147	0.224	0.27	0.135	0.275
there	0.135	0.044	0.137	0.182	0.045	0.171	0.223	0.147	0.206
this	0.331	0.132	0.308	0.183	0.107	0.175	0.211	0.129	0.206
two	0.354	0.146	0.319	0.249	0.078	0.252	0.131	0.083	0.136
type	0.294	0.104	0.27	0.098	0.098	0.088	-		
will	0.208	0.127	0.192	0.066	0.123	0	0.202	0.125	0.214
with	0.2	0.126	0.161	0.137	0.106	0.117	0.089	0.092	0.076
words	0.096	0.061	0.073	0.117	0.029	0.098	0.085	0.075	0.072

and highest for "are" = 0.285. With phone we make similar observations as that of tablet  $Dist_B$  values. All  $Dist_B$  values are less than 0.28. The least and highest mean values are for "words" = 0.085 and "the" = 0.27 respectively. We infer that StdFlight4 is not as discriminable as WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4. This feature does not provide improvement over the conventional features.

### 3.8.11 Analysis of proposed feature - StdKeyHold

**Description** : StdKeyHold is the standard deviation of all KeyHold values occurring within the context of a word.

**Inference** : Table 3.20 presents the mean  $Dist_B$  values for StdKeyHold on the three devices. The mean values of  $Dist_B$  for StdKeyHold are similar to those of all the conven-

Table 3.20: The Inter-User  $Dist_B$  values for StdKeyHold Distributions across all devices.

Device	Desktop			Tablet			Phone		
Word	Mean	StD	Median	Mean	StD	Median	Mean	StD	Median
are	0.497	0.183	0.495	0.507	0.152	0.508	0.454	0.165	0.479
carefully	0.225	0.062	0.244	-			0.245	0.283	0.133
data	0.289	0.143	0.283	0.501	0.133	0.524	0.398	0.156	0.397
different	0.291	0.267	0.172	0.355	0.185	0.367	-		
first	0.455	0.086	0.428	0.517	0.121	0.499	-		
have	0.277	0.121	0.286	0.31	0.183	0.319	0.394	0.223	0.452
lines	0.19	0.131	0.212	0.413	0.116	0.416	0.265	0.185	0.3
not	0.344	0.235	0.405	0.501	0.165	0.529	0.381	0.109	0.399
overlap	0.371	0.141	0.345	0.331	0.255	0.273	-		
phase	0.312	0.206	0.369	0.266	0.154	0.338	0.477	0.086	0.456
see	0.564	0.17	0.596	0.376	0.134	0.347	0.478	0.103	0.465
that	0.427	0.138	0.448	0.308	0.126	0.378	0.494	0.116	0.497
the	0.635	0.124	0.634	0.477	0.147	0.485	0.412	0.135	0.425
there	0.283	0.103	0.271	0.193	0.081	0.237	0.509	0.094	0.521
this	0.426	0.122	0.436	0.509	0.125	0.527	0.464	0.186	0.494
two	0.632	0.166	0.644	0.46	0.155	0.481	0.367	0.068	0.372
type	0.262	0.124	0.241	0.356	0.16	0.346	-		
will	0.222	0.215	0.085	0.378	0.147	0.324	0.323	0.179	0.335
with	0.335	0.176	0.351	0.394	0.137	0.387	0.307	0.234	0.329
words	0.255	0.095	0.26	0.371	0.061	0.392	0.435	0.119	0.445

tional features and are much inferior to those of all other proposed features on corresponding devices, but again for desktop they are similar to conventional feature values. Most mean  $Dist_B$  values for desktop lie around 0.2 to 0.5, which implies lesser separability. For tablet the mean  $Dist_B$  values are high, all values are above 0.26, and majority of them lying above 0.4. The least mean  $Dist_B$  value was for "phase" = 0.26 and highest for "first" = 0.517. This implies that this feature offers very less discriminability. With phone we make similar observations as that of tablet  $Dist_B$  values, all values lie above 0.24. The least mean  $Dist_B$  value is for "carefully" = 0.245. Overall, discriminability is low using this feature. All values on all devices suggest high overlaps in the PDFs, users are least separable using this among all the proposed features. Very few values on any de-

vice are desirable, and majority are high, therefore this feature is not a good feature for authentication/verification purposes.

We illustrate the discriminability of our proposed features using CDFs of  $Dist_B$  values as shown in Figure 3.5. The CDFs help us compare the separability of users, by each feature on all devices. These are the observations for each feature in comparison to conventional features and other proposed features. Figure 3.5a shows the CDF for WordHold. It appears to have higher discriminability in desktop about 60% of the samples having less than 0.4  $Dist_B$  and about 90% of them being less than 0.6  $Dist_B$ . On phone and tablet this feature is very good as almost 50% and 40% samples have 0  $Dist_B$  on phone and tablet respectively. 100% of the samples have below 0.35  $Dist_B$  which implies very good separation. We consider this feature to be evaluated in the next phase. Figure 3.5b shows the CDF for AvgFlight1. This shows an improved discriminability in desktop 60% of the samples have less than 0.3  $Dist_B$  and around 25% at 0. We also observe that  $Dist_B$  of 0.6 covers 90% of the samples. On phone and tablet this feature appears to be very good, as almost 35% and 25% of the samples have 0  $Dist_B$  on phone and tablet respectively and 100% below 0.4 which implies very good separation. This feature is considered for evaluation in the next phase.

Figure 3.5c shows the CDF for AvgFlight2. We observe that it is slightly less discriminative than WordHold in desktop, approximately 50% of the samples have less than 0.4  $Dist_B$  and 80% have less than 0.6. On phone and tablet this feature is very good, as almost 40% and 30% samples have 0  $Dist_B$  on phone and tablet respectively. Approximately 100% are below 0.4  $Dist_B$  which implies very good separation. We considered this feature for evaluation in the next phase. Figure 3.5d shows the CDF for AvgFlight3.

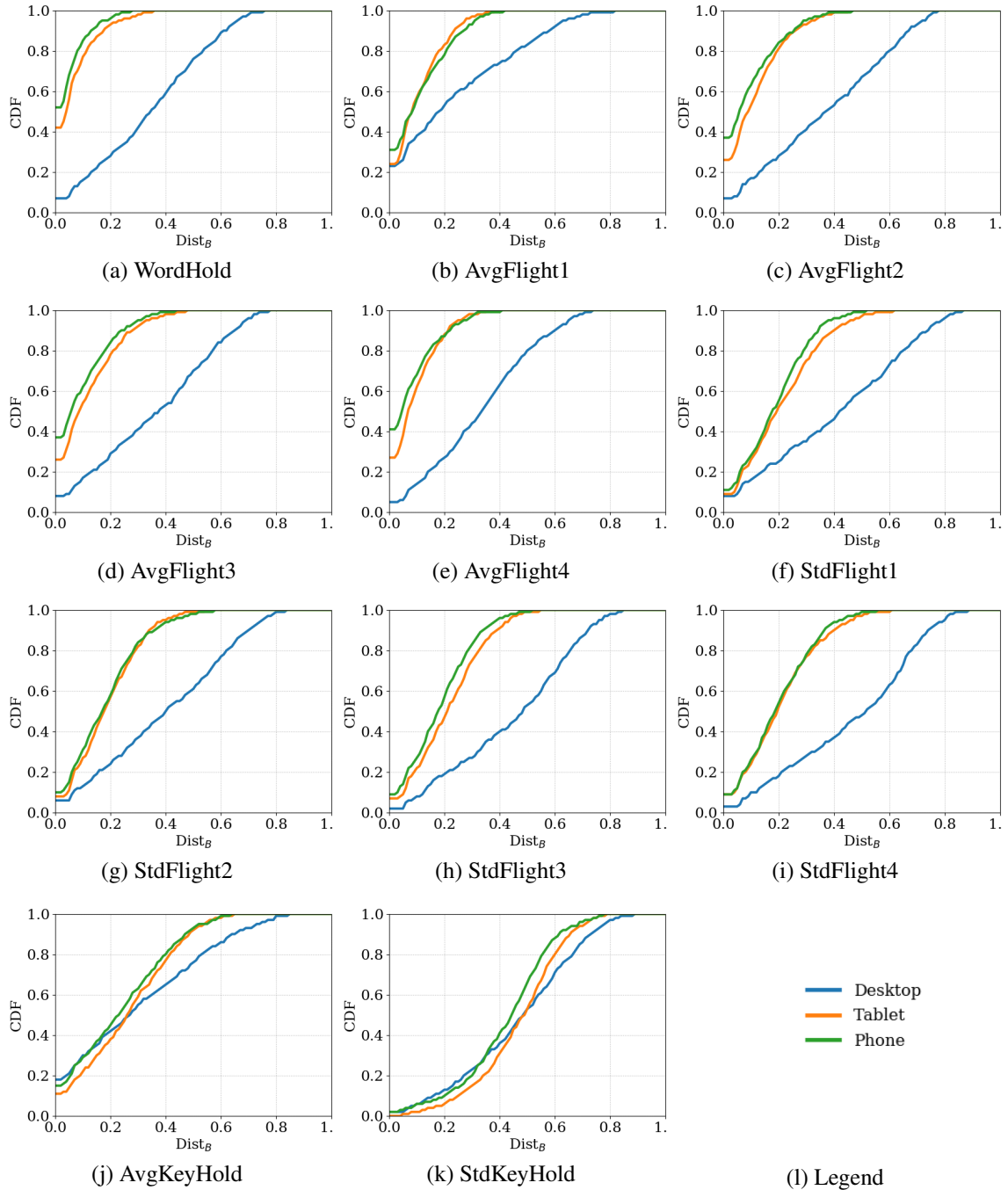


Fig. 3.5.: Comparing the Bhattacharyya distances of PDFs for all proposed context-sensitive features on desktop, tablet and phone.

Again, this feature is slightly less discriminative than WordHold in desktop approximately 50% of all samples have less than 0.4  $Dist_B$  and about 80% have less than 0.6. On phone and tablet this feature is very good, as almost 40% and 30% of samples have 0  $Dist_B$  on

phone and tablet respectively. Approximately 100% below 0.4  $Dist_B$  which implies very good separation. Feature is considered for evaluation in the next phase. Figure 3.5e shows that the CDF for AvgFlight4 is slightly more discriminative than AvgFlight3 with approximately 60% samples having less than 0.4  $Dist_B$  and 80% less than 0.6. On phone and tablet this feature has almost 40% and 30% of samples with 0  $Dist_B$  on phone and tablet respectively and 100% below 0.4  $Dist_B$  which implies very good separation. We will be considering this feature for evaluation in the next phase. Figure 3.5f shows the CDF for StdFlight1, For desktop about 60% samples have less than 0.5  $Dist_B$  and 80% less than 0.7 shows the feature is not very discriminative. On phone and tablet this feature is better than desktop, but not better than the other features discussed for phone and desktop. About 60% samples have below 0.3 and 90% have below 0.4  $Dist_B$ , not as good as 3.5a to 3.5e. We discard this proposed feature on grounds of not being discriminative enough. Figure 3.5g shows the CDF for StdFlight2 very similar to StdFlight1, for desktop about 60% of the values for  $Dist_B$  being less than 0.5 and about 80% of them being less than 0.7, implies that the feature is not very discriminative. On phone and tablet this feature is better than desktop, but not better than the other features discussed for phone and desktop. About 60% of the samples have below 0.3  $Dist_B$  and 90% have below 0.4 which is again not as good as 3.5a to 3.5e. We discard this feature too, as not being discriminative enough. Figure 3.5h shows the CDF for StdFlight3. We observe that for desktop, 60% of the samples have less than 0.6  $Dist_B$  and 80% have less than 0.7, which is not very discriminative. On phone and tablet this feature is better than desktop, but still have 60% of the  $Dist_B$  values below 0.3 and 90% below 0.4, which is not as good as 3.5a to 3.5e. We discard this feature as it does not offer improvement over conventional features. Figure

3.5i shows the CDF for StdFlight4 very similar to StdFlight3 , For desktop 60% values for  $Dist_B$  are less than 0.6  $Dist_B$  and 80% less than 0.7, not very discriminative. On phone and tablet this feature is better than desktop, but not better than the other features discussed for phone and desktop. 60%  $Dist_B$  values below 0.3 and 90% below 0.4, which is not as good as 3.5a to 3.5e. Feature is not considered for evaluation.

Figure 3.5j shows the CDF for AvgKeyHold slightly better values for desktop as about 60% of the samples have less than 0.4  $Dist_B$  and about 80% have less than 0.55. There seems to be marginal improvement compared to other proposed features. On phone and tablet this feature slightly better than desktop, but not better than the features we already discussed for phone and desktop. About 60% of the  $Dist_B$  values are below 0.3 and 90% are below 0.5 which is not as good as 3.5a to 3.5e. Feature is not considered for evaluation. Figure 3.5k shows the CDF for StdKeyHold and we observe that it is the least discriminative among proposed features. For desktop, about 60% of the  $Dist_B$  values are less than 0.6 and covers only 80% of the samples at a high  $Dist_B$  value of 0.7. This clearly implies that it is not very discriminative. Even in case of phone and tablet, 60% of. samples have values below 0.6 and 90% have  $Dist_B$  values below 0.7, which is not discriminative. This feature is not considered for evaluation.

### **Inference - Proposed Features**

We find that these features separate user keystroke data better than the conventional features. We also eliminate 6 of the proposed 11 features. Among the 11 proposed features: WordHold, AvgFlight1, AvgFlight2, AvgFlight3, AvgFlight4, StdFlight1, StdFlight2, Std-



Flight3, StdFlight4, AvgKeyHold and StdKeyHold. We find the most discriminative features to be: WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4. We use this subset of features for further evaluation, to build classifiers and see if these features can provide competitive accuracies for user identification, which is discussed in the next section.

### 3.9 Evaluation of proposed features

From the analysis on the user separability of both classes of features (Conventional and Proposed). It is clear that the proposed features offered higher separability. We further selected a subset of the proposed features which clearly had higher discriminative power like WordHold, AvgFlight1, AvgFlight2, AvgFlight3 and AvgFlight4 and discarded the rest of our proposed features. We tested these features by building classifiers based on them. We are using similar methodology as used by Sim and Janakiraman [163], [79]. To evaluate our proposed features and to measure the improvement in user identification achieved with them, we build classifiers with both, conventional features and proposed features. We then compare their results for the task of user identification. The underlying principle of the classifier is same for both the cases, the known histograms  $H_k$  of each person is compared to the histograms built from the input text  $H_i$ , We again use Bhattacharyya distance as described before, but this time we are looking for maximum overlap (max value between 0 to 1) between the two histograms. We take the average histogram overlap from all the features and identify the person as the one with the highest overlap. For conventional features we built classifiers for each of the digraphs discussed earlier.

For a digraph with characters  $d_i$  and  $d_{i+1}$  we use the argument maximize classifier represented by equation 3.2, as our classifier to identify the user.

Argument maximize classifier using conventional features for a digraph  $d_i d_{i+1}$ :

$$\begin{aligned}
 & \underset{u}{\operatorname{argmax}} \operatorname{Avg}(\operatorname{Dist}_B(H_{k d_i \text{KeyHold}}, H_{i d_i \text{KeyHold}}), \\
 & \quad \operatorname{Dist}_B(H_{k d_{i+1} \text{KeyHold}}, H_{i d_{i+1} \text{KeyHold}}), \\
 & \quad \operatorname{Dist}_B(H_{k d_i d_{i+1} \text{Flight1}}, H_{i d_i d_{i+1} \text{Flight1}}), \\
 & \quad \operatorname{Dist}_B(H_{k d_i d_{i+1} \text{Flight2}}, H_{i d_i d_{i+1} \text{Flight2}}), \\
 & \quad \operatorname{Dist}_B(H_{k d_i d_{i+1} \text{Flight3}}, H_{i d_i d_{i+1} \text{Flight3}}), \\
 & \quad \operatorname{Dist}_B(H_{k d_i d_{i+1} \text{Flight4}}, H_{i d_i d_{i+1} \text{Flight4}})).
 \end{aligned} \tag{3.2}$$

Argument maximize classifier using proposed features for a word  $X$ :

$$\begin{aligned}
 & \underset{u}{\operatorname{argmax}} \operatorname{Avg}(\operatorname{Dist}_B(H_{k X \text{WordHold}}, H_{i X \text{WordHold}}), \\
 & \quad \operatorname{Dist}_B(H_{k X \text{Avg-Flight1}}, H_{i X \text{AvgFlight1}}), \\
 & \quad \operatorname{Dist}_B(H_{k X \text{Avg-Flight2}}, H_{i X \text{Avg-Flight2}}), \\
 & \quad \operatorname{Dist}_B(H_{k X \text{Avg-Flight3}}, H_{i X \text{Avg-Flight3}}), \\
 & \quad \operatorname{Dist}_B(H_{k X \text{Avg-Flight4}}, H_{i X \text{Avg-Flight4}})).
 \end{aligned} \tag{3.3}$$

For the proposed features we used a argument maximize classifier represented by equation 3.3 as our classifier for a word  $X$ . Using the most common unigraphs and digraphs discussed in the previous sections, we built 11 classifiers that used the conventional features and 20 classifiers, one for each word considered in previous section which used our

proposed features. We used a Synthetic Minority Over-sampling Technique (SMOTE), as described in Chawla et al. [40], to balance and oversample our data when needed, and use 10-fold cross validation and report the mean and standard deviation of the accuracies of our classifiers. As our data was balanced and we had 20 users, the chance of random guess identification was 0.05. Tables 3.21 and 3.22 present the accuracies of all the classifiers. In case of the conventional features, we can observe low accuracies in most cases. Except for a few digraphs in case of hand held devices ((space,i), (space,a), (space,s)), accuracies range from 45% to 75%, which is not very desirable. The accuracy of classifiers are extremely low in case of the desktop with conventional features, with the highest accuracy being 68% for the digraph (space,i). Whereas, while using the classifiers with proposed subset of features we see that most accuracies lie in the range of 87% to 97%. With desktop we found that classifiers for the words: "data", "first", "have", "that" had accuracies of over 90%, while many others performed fairly well with accuracies above 85%, even the worst performer: "type" was 73.2% accurate. Both hand-held devices have high accuracies for a majority of the words selected. On tablet the classifiers built for words: "see", "that", "there", "with" had accuracies of 95% and above, lowest accuracies was for the word "two" at 75.3%. On phone the classifiers built for words: "data", "have", "see", "this", "with" had accuracies of above 93%. The lowest accuracy on phone was for the classifier of the word "phase" at 85%.

Table 3.21: Classifier accuracies for the conventional feature based classifiers in our experiment.

Device	Desktop		Tablet		Phone	
	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.
(`space`, `a`)	0.63	0.04	0.76	0.12	0.8	0.097
(`space`, `i`)	0.68	0.04	0.81	0.163	0.83	0.08
(`space`, `s`)	0.58	0.053	0.79	0.091	0.8	0.105
(`space`, `t`)	0.42	0.07	0.69	0.053	0.76	0.093
(`e`, `space`)	0.44	0.032	0.59	0.06	0.67	0.075
(`e`, `r`)	0.55	0.035	0.66	0.055	0.7	0.064
(`e`, `s`)	0.52	0.071	0.62	0.087	0.68	0.077
(`o`, `n`)	0.69	0.045	0.67	0.044	0.72	0.069
(`r`, `e`)	0.53	0.06	0.58	0.053	0.59	0.062
(`t`, `space`)	0.56	0.063	0.62	0.08	0.71	0.092
(`t`, `h`)	0.49	0.03	0.58	0.035	0.66	0.056

Table 3.22: Classifier accuracies for the proposed feature based classifiers in our experiment.

Device	Desktop		Tablet		Phone	
	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.
are	0.85	0.071	0.87	0.056	0.91	0.054
carefully	0.86	0.063	-	-	0.89	0.051
data	0.93	0.056	0.915	0.055	0.975	0.034
different	0.88	0.046	0.875	0.06	-	-
first	0.93	0.033	0.84	0.073	-	-
have	0.905	0.035	0.91	0.07	0.963	0.034
lines	0.875	0.046	0.85	0.036	0.91	0.044
not	0.895	0.057	0.91	0.049	0.915	0.05
overlap	0.875	0.06	0.868	0.049	-	-
phase	0.875	0.068	0.84	0.03	0.859	0.06
see	0.895	0.035	0.96	0.037	0.935	0.039
that	0.905	0.072	0.95	0.039	0.911	0.053
the	0.785	0.045	0.865	0.055	0.932	0.041
there	0.895	0.052	0.963	0.034	0.911	0.075
this	0.889	0.05	0.875	0.068	0.945	0.042
two	0.86	0.037	0.753	0.058	0.865	0.059
type	0.732	0.055	0.821	0.042	-	-
will	0.885	0.071	0.87	0.046	0.889	0.064
with	0.89	0.07	0.95	0.039	0.935	0.055
words	0.868	0.063	0.86	0.062	0.863	0.059

### 3.10 Insights drawn from the analysis on proposed features

We show that the conventional features have lower discriminative power between users when compared to the proposed context sensitive features, this leads to many more important and intriguing questions;

#### 3.10.1 Insight 1: Why do the proposed features perform better than the conventional features?

To answer this question, we use the analysis of Entropy, as a measure of disorder in data, and explain why proposed features tend to perform better than the conventional features. To be more precise, the information gain, which is a measure of the decrease in disorder that is achieved by partitioning the data is the key concept being exploited with the help of our proposed features. Let  $x$  be a unigraph or a digraph. In the conventional approach, features values associated with  $x$  (*Keyhold* if  $x$  is a unigraph, *flight<sub>1</sub>* through *flight<sub>4</sub>* if  $x$  is a digraph) are grouped together irrespective of context. Let  $\vec{d}_x$  represent vector of feature values extracted for  $x$  from text sample. The entropy of  $\vec{d}_x$  can be expressed as:

$$E(\vec{d}_x) = - \sum_{i=1}^k p_i \log_2 (p_i), \quad (3.4)$$

Where  $i = 1$  to  $k$  are the number of bins into which  $\vec{d}_x$  is split and  $p_i$  is the probability of the feature value of  $x$  being in bin  $i$ . Practically, the values forming  $\vec{d}_x$  come from different contexts or words as described in previous sections. Let  $\vec{w}$  represent the vector of different words from which the values of  $\vec{d}_x$  are extracted. Therefore, if  $\vec{d}_x$  were to be

partitioned based on the context from which its values come from, the entropy of such partition can be expressed as:

$$E(\vec{d}_x, \vec{w}) = \sum_{j=1}^m \left( \frac{n_j}{n} x E(\vec{d}_{x_j}) \right), \quad (3.5)$$

Where  $j = 1$  to  $m$  are the number of different words that values of  $x$  are extracted from. Essentially, equation (3.5) gives us the sum of the weighted average of the entropy from all the partitions. The information gained by performing this partition or in other words, the reduction in disorder achieved by the partitioning can be expressed as:

$$I(\vec{d}_x, \vec{w}) = E(\vec{d}_x) - E(\vec{d}_x, \vec{w}). \quad (3.6)$$

Therefore, theoretically,  $I(\vec{d}_x, \vec{w}) \geq 0$ , but, practically, the cases where  $I(\vec{d}_x, \vec{w}) = 0$  can only occur if  $\vec{w}$  partitions  $\vec{d}_x$  into partitions with the same probabilities as  $\vec{d}_x$ . Since  $\vec{w}$  inherently consists of only high frequency words, the chances of  $E(\vec{d}_x) = E(\vec{d}_x, \vec{w})$  are not practical and hence  $I(\vec{d}_x, \vec{w}) = 0$  is highly unlikely. Therefore, however small the difference between  $E(\vec{d}_x)$  and  $E(\vec{d}_x, \vec{w})$  maybe, it is highly likely to lead to a positive information gain, or reduction of disorder in data, which in our case are the feature values. Reduction in disorder means the partitions have much more homogeneous values than when taken without the partition, which intuitively should be better for classification or identification. We observed that entropy of the feature values when extracted with context restrictions, were much lower than a global extraction approach. Because  $I(\vec{d}_x, \vec{w}) \geq 0$ ,

and the case of  $I(\vec{d}_x, \vec{w}) = 0$  is impractical, this answers our question about why the proposed features work better than the conventional features.

Table 3.23: Example from our desktop dataset: average feature values for a randomly chosen user shows the variations in the average feature values for the character "h" and digraph "ha" depending on the context in which they appear. All values are in milliseconds.

	Avg. KeyHold time of "h"	Avg. [Flight1, Flight2, Flight3, Flight4] time of "ha"
Over the entire sample	183.45	[46.6, 270.2, 241.2, 470.0]
Over all occurrences of "that"	167.20	[12.4, 230.0, 201.3, 389.5]
Over all occurrences of "have"	189.15	[44.7, 279.6, 252.0, 461.8]

An example of how the feature values vary with respect to context is shown in Table 3.23 where we chose a random user from our dataset and calculated the average feature values for character "h" and digraph "ha", first, over all occurrences, then, only over all occurrences in the words "that" and "have" separately. When computed overall occurrences, the average key-hold time for "h" was about 183.5ms which is much higher than key-hold time 167.2ms, in all occurrences of "that". The average flight1 values of "ha" for all occurrences of "that" and "have" were 12.4ms and 44.7ms respectively. Similar distinction can be found in all feature values as shown in Table 3.23, it is representative of a majority of the dataset for all users. From this example, it is clear that context affects the feature values.

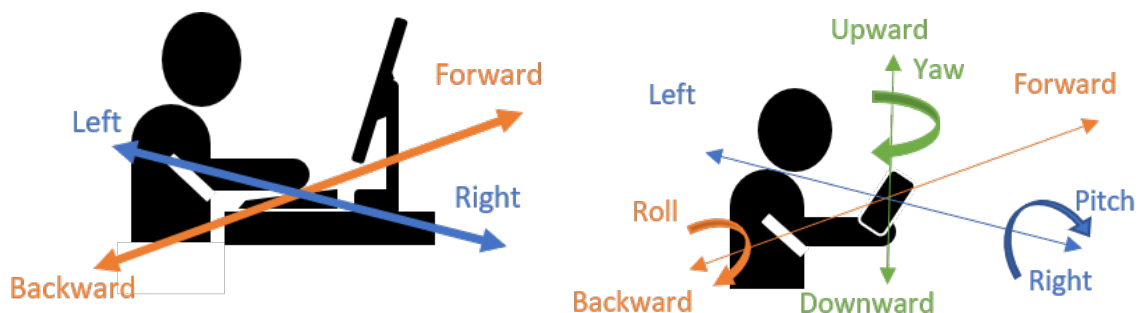
### 3.10.2 Insight 2: Why are the results and the performance of proposed features better in case of hand-held devices when compared to the desktop?

To help answer this question we look for inherent differences between the devices, their usage and the data collected on them. There are several reasons that we suspect, might

lead to the difference in the efficiency of the features. First, the very nature of the usage of these devices are different. As shown in Figure 3.6a, a typical keyboard when used on a table allows only two degrees of freedom, forward/backward and left/right. In comparison, a mobile phone offers six degrees of freedom; forward/backward, left/right, upward/downward, yaw, pitch and roll as shown in Figure 3.6b. This can affect the style of using a keyboard and hence bring in some differences in data collected from these families of devices.

Another reason for this difference in effectiveness of features, we suspect, is the consistency in the typing speeds. We found that a majority of the users had larger standard deviations in the time required to type a complete word on the desktop when compared to the hand-held devices. In particular, our proposed feature "*WordHold*" reflected this behavior for most of the users. The standard deviation for *WordHold* of a word on the hand-held devices was about  $3/4^{th}$  that of the standard deviation of *WordHold* of the same word on desktop typed by the same user. We also observed that the *KeyHold* duration for "space" between words and the *Flight1* duration from "space" to the first letter of words, were more densely clustered in case of the desktop when compared to the hand-held devices. On hand-held devices, this behavior gave the impression of small uniform bursts of typing activity(word) followed by nonuniform pauses(space). In contrast typing behavior on desktops appeared to be nonuniform throughout. As all users in our study owned





(a) Degrees of freedom while typing on a desktop. (b) Degrees of freedom while typing on a phone.  
 Fig. 3.6.: Typically, a desktop keyboard offers only two degrees of freedom; forward/backward and left/right. As a typical phone can be held by its user in any comfortable posture, it offers six degrees of freedom; forward/backward, left/right, upward/downward, yaw, pitch and roll as shown in these figures.

smart-phones and indicated that their usage of phones was much higher than their usage of desktops, we posit that it leads to these typing patterns on different devices.

### 3.10.3 Insight 3: What word-based factors might impact the user identification performance of proposed features?

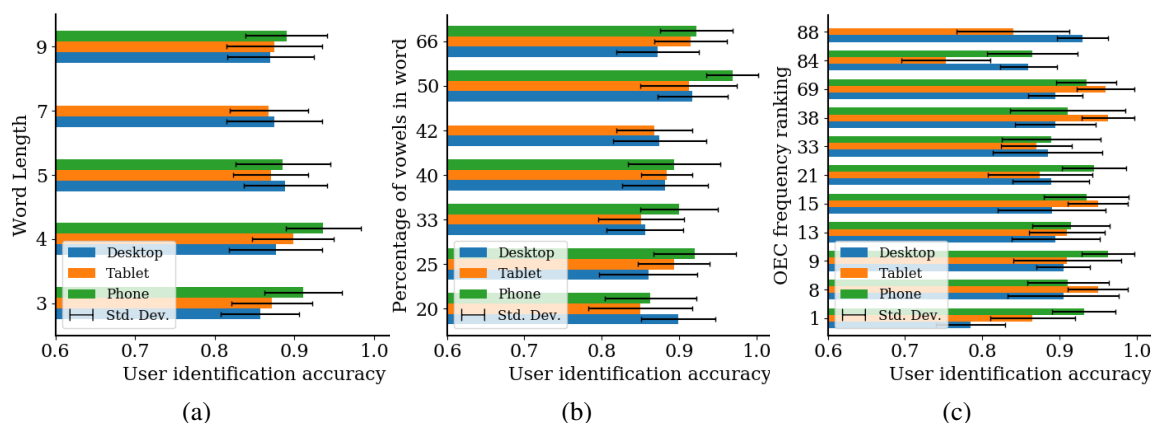


Fig. 3.7.: Impact of three word-based factors on the performance of proposed features for user identification. The three factors are: Word length (Fig. 3.7a): number of characters in a word; Vowel Percentage (Fig. 3.7b): percentage of vowels in a word; and Oxford English Corpus (OEC) frequency ranking (Fig. 3.7c): the frequency ranking of the words in our study according to OEC (Top 100). The words in order of rank (Fig. 3.7c, y-axis) are: (1, the), (8, that), (9, have), (13, not), (15, with), (21, this), (33, will), (38, there), (69, see), (84, two) and (88, first).

Figure 3.7 shows the impact of three word-based factors: (1) word length; (2) percentage of vowels in the word; and (3) Oxford English Corpus (OEC) frequency ranking (top 100)[141], on the performance of the proposed features for user identification (Table 3.22). Short words (3 to 4 characters) performed slightly better on hand-held devices whereas medium length words (5 to 7 characters) performed slightly better on the desktop as shown in Figure 3.7a. For the percentage of vowels in a word we observe a gradual improvement in accuracy until 50% of the word is comprised of vowels (Figure 3.7b). For all 3 types of devices, the identification accuracy peaks when the words to extract features have 50% vowels. The ranking (Figure 3.7c) consists of words in our study which are also most frequent (top 100) in Oxford English Corpus (OEC)[141] and the Corpus of Contemporary American English (COCA)[51]. The medium length words (4 to 5 characters), words with about 50% vowel composition, and those that are ranked higher in frequency give the best results for user identification using the proposed features. The word "the" is an exception to these observations because it performs poorly on the desktop and tablet even though it is the most used word in the corpora.

#### 3.10.4 Discussion: Attacks and limitations

The most common types of attacks on text in literature are: (1) for inferring typed text (generally PIN and Password through side channel attacks) or (2) for inferring keystroke timings (generally impersonation attacks). We discuss both in the following paragraphs.

**Inferring typed text.** Side channel attacks, such as video ([41], [161]), smartwatch [109], acoustic signals ([107], vibrations in video [80] and Channel State Information ([6],[57])

use some form of eavesdropping to obtain the victim's typed text rather than impersonating their typing behavior. The domain of these attacks does not address keystroke timings so they are not applicable to our scenarios.

**Inferring keystroke timings.** Attacks to mimic the keystroke timings of a victim are forms presentation attacks, where typically the attacker formulates an imposter text sample (including the keystroke timings) from data drawn from the statistics of acquired samples ([142], [155]), either stolen or from publicly available databases. Typically, these attacks use conventional features and assume the latencies to be similar across the entire keystroke data; however, our analysis and examples (Table 3.23) show that keystroke latencies vary depending on the context that the keys appear in. Though similar methods could be applied to attack our proposed features, the obstacles to factor in the context of keys would be significant. Additionally, Stefan and Yao [166] and Huang et al. [77] also provide measures to defend against such attacks. Another attack presented by Khan et al. [86] explored Augmented Reality (AR) to mimic a user's typing with assistance from the AR system. Apart from using conventional features, the attack made several non-trivial assumptions regarding the availability of victim's device and keystroke timings. The settings of this attack are complex and difficult in real life scenarios; however, applicability, although unlikely to our proposed features, needs exploration. The viability of an attack

when an attacker obtains both, the typed text and the keystroke timings, is an open problem that needs exploration.

### **3.10.5 Conclusion and future work**

We show that proposed word-specific features perform much better at user identification on all devices. Conventional features, especially KeyHold does not provide user separation to a desired level. We considered the subset of proposed features that offered higher discriminability, like WordHold, AvgFlight1, AvgFlight2, AvgFlight3, AvgFlight4, evaluated them with classifiers and drew comparisons with conventional features (Section 3.9). These classifiers show competitive accuracies on all devices. Mathematical insights for this improvement in performance are drawn (Section 3.10.1). We also note that these features in general perform much better on hand-held devices. We speculate that user's style of holding devices and patterns such as, short bursts of typing followed by pauses between words might be some of the reasons (Section 3.10.2). Analysis of the word-based impact factors reveal that four or five character words, words with about 50% vowels, and those that are ranked higher on the frequency lists might give better results for the extraction and use of the proposed features (Section 3.10.3) for user identification.

The results of our experiment call for a shift from conventional features to word specific features for continuous authentication using KD. We are of the opinion that factoring in the knowledge of context can be beneficial to KD. We hope this article provides direction to researchers of optimum KD features. As part of our future work, we are exploring inter-device relationships in KD, to see how a user's behavior on a device is linked to their behavior on another.

#### 4. BEHAVIORAL BIOMETRICS : THE FAILURE OF NORMALITY ASSUMPTION

A common assumption in behavioral biometrics, is that feature values follow a normal distribution. This assumption impacts key facets of research such as; decisions of sampling techniques and authentication models; and performance and results from the resulting systems. Our work raises the questions, "Should the *assumption of normality* be the *norm* in behavioral biometrics?" and "How normal is the assumption of normality?". We posit that our results will *change* how classification is approached by emphasizing the validity of assumptions about the underlying data. This work has the potential to impact a large body of work in behavioral biometrics.

Behavioral biometrics are ideal for continuous authentication on devices as the data for authentication can be acquired from user's regular activity without interrupting them to provide separate test samples. Many researchers assume the underlying distribution of features from these behavioral biometrics to be normal ([164, 167]). Such assumptions are made for theoretical or calculational simplicity discussed in later sections. But, the effects of assuming an underlying normal distribution in data when it is not, have been explored in various fields. This misassumption can be a source of error in classifications, when it influences the decision rules in the classifier, and in an adversarial approach lead to inefficient attacks if the generative model depends only on the mean and standard deviation.

We use our benchmark SU-AIS BB-MAS [26] dataset, with the data from 117 users providing keystrokes data on desktop, phone and tablet; accelerometer and gyroscope data while walking, upstairs and downstairs while carrying phone and tablet; and touch screen swiping data on phone and tablet; for our experiments. We performed a large number of Lilliefors test and Shapiro-Wilk's test on all modalities in our dataset. To summarize results, we categorize the features into four different categories such as; a) less than 25% of samples with p-value  $>0.05$ ; b) 25% to 50% of samples with p-value  $>0.05$ ; c) 50% to 75% of samples with p-value  $>0.05$ ; and, d) above 75% samples with p-value  $>0.05$ . We find that except for the features from samples taken from climbing upstairs and downstairs data, almost all features from samples of all other activities were in the category where less than 25% of samples warranted the null hypothesis be not discarded. Although many other fields have witnessed research work cautioning the naivety in the assumption of normality, in behavioral biometrics, this assumption has hardly been examined. Lesser so in the case of multiple activities that span multiple devices. We present related literature, briefly explore the reasons why researchers assume normality in data, describe the SU-AIS BB-MAS dataset and various extracted features, explain our experiments in detail and conclude with the impacts of our findings.

#### **4.1 Key contributions of the chapter**

- Experiment on a large dataset with 117 users providing; 3.5 million keystroke events; 57.1 million data-points for accelerometer and gyroscope each; and 1.7 million data-points for swipes. Each data session is between 2 to 2.5 hours each, consist-

ing multiple activities such as: typing (free and fixed text), gait (walking, upstairs and downstairs) and swiping activities while using desktop, phone and tablet. Thus, making our experiments and insights encompassing for most common behavioral biometrics

- Most features that are commonly described in literature, for data from all activities in the dataset have been extracted and examined for an underlying normal distribution using suitable non-parametric tests.
- Discuss alternate approaches that researchers may explore to mitigate or avoid the effects of assuming a non-existent normal distribution in data.
- Present implications of our findings for future work, such as the considerations for modelling distributions and classifier choices.

## 4.2 Related work

Behavioral biometrics includes a broad spectrum of modalities involving human behavior while performing day-to-day tasks. Keystrokes, Gait and Swiping Patterns are the most explored behavioral biometrics in recent times. Researchers have focused on utilizing various aspects of these behavioral biometrics for authentication ([15, 90]), verification [122], continuous authentication [117], gender detection ([36, 177]), age detection [138], fatigue detection [178], mood disturbance detection [198], lie detection [115] and detection of various health conditions. In many of these works, researchers have either purposefully or inadvertently, assumed a underlying normal distribution in the features

extracted from the data, reflected in their methods of analysis or in their choice of classifiers. Although assuming a normal distribution helps in simplifying the problem, it may not always lead to the best results.

The adverse effects of assuming a normal distribution has been studied greatly in different fields. In the field of Constraint Satisfaction Problems (CSP), [92] proves that in the results produced by many heuristic combinations on random binary CSPs and 3-colouring problems, the benchmarks for CSP, the assumption of normality does not hold. The authors also appeal for statistics that do not rely on the normality assumption to analyze empirical results for CSP. In processes involving classification of remotely sensed data from different spectral bands (image classification is a subset of this problem family), Olson [133] showed that the brightness values distributions did not follow a normal distribution. They further remarked and this fallacy was a major source of error in land cover classification when decision rules employed in the classifier assumed an underlying normal distribution. In Dunning's [55] work on the statistical analysis of text, they pointed out that the assumption of normal distribution limits the ability to analyze rare events and that those rare events were a large fraction of real text.

When dealing with the solution space for economic design of X-bar control charts, [38] shows that non-normality assumption also has a more significant effect on the Type II error probability than the Type I error probability. In the research of human cancer genomes, wrongly assuming a normally distributed Gene expression was shown to affect multiple facets, including identification of expression patterns, annotation and classification [110]. They also concluded that small departures from normality were not analytically insignificant. Limpert and Stahel [102], questioned the adequacy of characterization of data using



normal distributions and argue that an asymmetric view will increase, recognition of data distributions and also the quality of interpretation.

#### 4.2.1 The Central Limit Theorem and Cràmer-Rao Lower Bound

We found two main concepts that many researchers use to justify their assumption of normality in most datasets; the Central Limit Theorem (CLT) [91] and the Cràmer-Rao Lower Bound (CRLB) [49].

The popularity in the assumption of normality can be attributed to the fact that noise in many systems has been represented well using a normal distribution. Gaussian assumption is a good conservative choice when not much is known about the data, which is also supported by CLT stating that the distribution of sample means tends to form a normal distribution as the sample size gets larger. A Gaussian distribution also minimizes the Fisher Information, which is the inverse of CRLB. In other words, the CRLB under the Gaussian distribution works for the worst-case scenario, maximizing the CRLB (see [170]). Therefore, minimizing the largest CRLB is interpreted as min-max optimal [135].

#### Central Limit Theorem

If  $X_1, X_2, \dots, X_n$  is a random sample of size  $n$  taken from a population with mean  $\mu$  and finite variance  $\sigma^2$ , and if  $\bar{X}$  is the sample mean, the limiting form of the distribution of

$$Z = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}} \quad (4.1)$$

as  $n \rightarrow \infty$ , is the standard normal distribution (see Theorem 7.2 in [123]).

Researchers often assume that estimated population mean and variance are independent for simplicity, implying sample variance means and variance are independent, which is true only for normal distribution.

Many research works inadvertently fall back on CLT for their modelling choices. A good example of this scenario is the use of Gaussian Mixture models for keystroke analysis ([30, 74, 199]) and touchscreen swipe analysis ([60, 140]). A Gaussian Mixture model assumes the data to follow a mixture of individual multivariate Gaussians or Gaussian Mixture distribution. Which is only a good choice if the data is a mixture of Gaussians and has large enough number of samples, implicitly invoking CLT.

### **Fisher Information and Cràmer-Rao Lower Bound**

Fisher information for a random sample from  $f(x|\theta)$ , where  $\theta$  is an unknown parameter and  $n \rightarrow \infty$ , is expressed as

$$I(\theta) = -E \left[ \frac{\partial^2}{\partial \theta^2} \log f(X|\theta) \right] \quad (4.2)$$

(see Theorem 5.8 [96])

If  $\hat{\theta}$  is an unbiased estimator of  $\theta$ , and the Cràmer-Rao Lower Bound [49] states that the variance of  $\hat{\theta}$  is bounded by the reciprocal of Fisher Information, i.e.

$$\text{var}(\hat{\theta}) \geq \frac{1}{nI(\theta)} \quad (4.3)$$

using CLT, it follows that as sample size tends to infinity, the maximum likelihood estimator is asymptotically unbiased and asymptotic distribution of  $\hat{\theta}$  is normal. i.e. for a true value  $\theta_0$  for  $\theta$

$$\sqrt{nI(\theta_0)}(\hat{\theta} - \theta_0) \sim N(0, 1) \quad (4.4)$$

The Gaussian assumption is often applied in modelling of keystroke latencies. For example, Song et al., in [164], modelled the latencies of 142 pairs of characters under the assumption that, "probability of the latency  $y$  between two keystrokes of a character pair forms a univariate Gaussian distribution" and thereby deriving the  $\mu$  and  $\sigma$  parameters for each character pair, to be used further down for information gain estimation. Similarly, Stefan et al., in [167], begin their bot simulations under the assumption that the keystroke duration of a character in a word is modeled as a random variable which is Gaussian or a constant with additive uniform noise. Thus, their typing event injections for synthetic forgeries or the bot attacks are already under assumptions that should not be made casually.

#### 4.2.2 Related work in keystroke dynamics

In [154], the authors examined keystroke features such as, Key Interval Times and Key Hold Times extracted from keystroke data recorded on desktops, to test if they followed a normal distribution. The authors performed Lilliefors [100] and Cramer-von Mises [168] tests and established that Key Hold Times and the Key Interval Times from desktop typing did not follow a Gaussian distribution for all Unigraphs and Digraphs. As research in keystrokes dynamics is a popular component of behavioral biometrics, this can be con-

sidered as a pioneering step towards questioning the normality assumption in behavioral biometrics.

### **How our work differs from related work discussed**

Although many other fields have questioned the naivety of the assumption of normality, the field of behavioral biometrics has been lagging in this aspect. We found that a section (5.1) of the research in [154], was the only exploration in this direction for keystroke dynamics on desktops. Behavioral biometrics is an emerging area with many different modalities, devices and activities. For example; typing is no more limited to a desktop, keystroke dynamics have been studied separately on phones and tablets and other touch devices; gait as a behavioral biometric has been studied with different sensor placements, different sub-activities and different devices; and, swiping patterns for behavioral biometrics have been studied on different smart-screens or touch surfaces.

We examine the assumption of normality, in a large dataset consisting of a wide range of activities and devices. By performing a large number of statistical tests, we show the following:

- Keystroke features such as keyhold and flight times, do not follow normal distribution on desktops, tablets and phones. Our experiments reaffirm the findings in [154] that showed the non-normal nature of keystroke features on desktop. We observe the same occurrence in keystroke features from phone and tablets, which have not been examined in literature.

- Features extracted from accelerometers and gyroscopes of tablets and phones (in pocket and in hand), do not follow a Gaussian distribution while walking. However, in a large number of samples from activities of climbing up and down the stairs, the null hypothesis that the data comes from normal distribution cannot be discarded.
- Swipe features extracted from swipe trajectories, pressure, acceleration and touch area data do not follow normal distribution on smart-phone and tablet surfaces.
- We also discuss the implications and alternate approaches for non-normal distributions in data.

### **4.3 Data and features**

In this section, we briefly describe the dataset and the features that we have analyzed. We have considered the most popularly used features from each of the modalities.

#### **4.3.1 Details of the dataset**

For all the experiments and analysis described in this article, we use our open-access benchmark dataset SU-AIS BB-MAS [26]. In [25], we describe all aspects of the dataset in great detail. Therefore, we only provide a gist of it in this section. A total of 117 users participated in the voluntary data collection which was carried out after the IRB approval from our university. The dataset consists a total of about: 3.5 million keystroke events; 57.1 million data-points for accelerometer and gyroscope each; 1.7 million data-points for

swipes; and enables future research to explore previously unexplored directions in inter-device and inter-modality biometrics.

Table 4.1: The different types of data, from SU-AIS BB-MAS [26], that we analyzed from multiple devices and activities. The gait activity consists of three sub-activities, walking, climbing upstairs and downstairs.

Activity	Devices	Data/Sensor
Typing	Desktop Phone Tablet	Keystroke Timings
Gait	Phone in Pocket Phone in Hand Tablet in Hand	Accelerometer Gyroscope
Swipes	Phone Tablet	Touchscreen

Table 4.2: List of features extracted and examined in our experiments for an underlying normal distribution.

Data	Features	Details
Typing	<ul style="list-style-type: none"> <li>•Keyhold</li> <li>•Flight 1 to Flight 4</li> </ul>	<ul style="list-style-type: none"> <li>–Keyhold times were extracted from twelve most occurring unigraphs</li> <li>–Flight times were extracted from eighteen most occurring digraphs</li> </ul>
Gait	<ul style="list-style-type: none"> <li>•Mean</li> <li>•Standard Deviation</li> <li>•Band Power</li> <li>•Energy</li> <li>•Median</li> <li>•Inter Quartile Range</li> <li>•Range</li> <li>•Signal to Noise Ratio</li> <li>•Dynamic Time Warp Distance</li> <li>•Mutual Information</li> <li>•Correlation</li> </ul>	<ul style="list-style-type: none"> <li>–Features were extracted from <math>x, y, z</math> and <math>m</math> where <math>(m = \sqrt{x^2 + y^2 + z^2})</math> signals from both the accelerometer and gyroscope.</li> <li>–All features were extracted from each of the directional signals except for DTW distance, Mutual Information and Correlation which were extracted between pairs of these signals i.e., <math>x-y, x-z, x-m, y-z, y-m</math> and <math>z-m</math>.</li> </ul>
Swipe	<ul style="list-style-type: none"> <li>•Minimum <math>x</math> and <math>y</math> coordinates</li> <li>•Maximum <math>x</math> and <math>y</math> coordinates</li> <li>•Euclidean Distance</li> <li>•Angle of the swipe</li> <li>•Time</li> <li>•Velocity Mean and Std.</li> <li>•Velocity Quartiles</li> <li>•Acceleration Mean and Std.</li> <li>•Acceleration Quartiles</li> <li>•Pressure Mean and Std.</li> <li>•Pressure Quartiles</li> <li>•Area Mean and Std.</li> <li>•Area Quartiles</li> <li>•Direction</li> </ul>	<ul style="list-style-type: none"> <li>–Features were extracted from a variety of information making up a swipe.</li> <li>–Features such as coordinates, angles and direction are dependent on the touch points on the screen and the end points of a swipe.</li> <li>–Velocity, Pressure, Acceleration and Area are calculated with the data from corresponding sensors on the touch surface of the devices.</li> <li>–The Direction feature was only used to group the swipes into vertical and horizontal swipes</li> </ul>

### 4.3.2 Details of the features

We extracted popular features that have been used in literature for each modality. The feature extraction for our dataset can be grouped into three parts, namely keystroke, gait and swipe features. We briefly describe the features and their storage below. A summary of the features is presented in the Table 4.2.

- **Keystroke Features:** We select the most occurring twelve unigraphs (single key) and eighteen digraphs (pair of consecutive keys) that occurred the most number of times in all user's keystroke data. The unigraphs are : "BACKSPACE", "SPACE", "a", "e", "h", "i", "l", "n", "r", "S" and "t". The digraphs are: ('BACKSPACE', 'BACKSPACE'), ('SPACE', 'a'), ('SPACE', 'i'), ('SPACE', 's'), ('SPACE', 't'), ('e', 'SPACE'), ('e', 'n'), ('e', 'r'), ('e', 's'), ('n', 'SPACE'), ('o', 'SPACE'), ('o', 'n'), ('r', 'e'), ('s', 'SPACE'), ('s', 'e'), ('t', 'SPACE'), ('t', 'e') and ('t', 'h'). For a unigraph  $K_i$  we extract the *Keyhold* time of the key as a feature:

$$- \text{Keyhold}_{K_i} : K_i \text{Release} - K_i \text{Press}$$

For a digraph  $K_i$  and  $K_{i+1}$  the following temporal features are extracted:

$$- \text{Flight1}_{K_i K_{i+1}} : K_{i+1} \text{Press} - K_i \text{Release}$$

$$- \text{Flight2}_{K_i K_{i+1}} : K_{i+1} \text{Release} - K_i \text{Release}$$

$$- \text{Flight3}_{K_i K_{i+1}} : K_{i+1} \text{Press} - K_i \text{Press}$$

$$- \text{Flight4}_{K_i K_{i+1}} : K_{i+1} \text{Release} - K_i \text{Press}$$

**Outlier removal for Keystroke Features:** We use a simple filter to remove any instances of keys that were held down for two seconds or more. We also remove instances

of the inter-key pauses that are greater than two seconds. We assume that these were caused by pauses, where the user is either thinking or receiving instructions during the data collection.

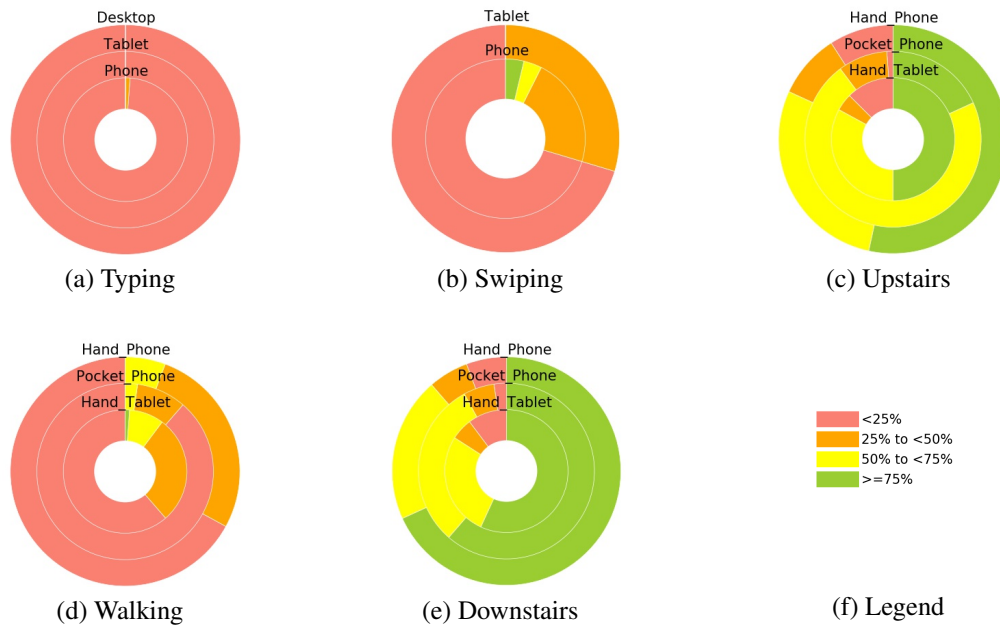


Fig. 4.1.: Illustration summarizing the amount of features in each activity and percentage of their samples with  $p > 0.05$ , or in other words, where the null hypothesis  $H_0$ , that the samples came from a normal distribution could not be discarded. The categories and their corresponding color codes are; **red**- less than 25% of samples with p-value  $>0.05$ ; **orange**- 25% to 50% of samples with p-value  $>0.05$ ; **yellow**- 50% to 75% of samples with p-value  $>0.05$ ; and, **green**- above 75% samples with p-value  $>0.05$ . A full doughnut in the doughnut chart represents all the features for an activity on the labelled device. For example, the outer most doughnut in Figure 4.1a, represents all the features examined for keystrokes latencies on desktop, the second doughnut for tablet and innermost doughnut for phone respectively. The area covered by each color/category on a doughnut represents the amount of features that fall in the color/category as described above.

- Gait Features:** As the raw data for the gait is a pair of signals from the accelerometer and gyroscope we extract features from both. The gait data is further subdivided into three activities; "Walking" (on a flat corridor); "Downstairs" (going down the staircase); and, "Upstairs" (going up the staircase). We use a window size of two seconds with a one second overlap between two consecutive windows. For each two second



window we extract a host of features from the accelerometer and the gyroscope for x, y, z and m ( $m = \sqrt{x^2 + y^2 + z^2}$ ). The list of features extracted is given in Table 4.2. While Mean, Standard deviation, Band power, Energy, Median frequency, Inter quartile range, Range, Signal to noise ratio are extracted for x, y, z and m, Dynamic time warping distance is calculated only between pairs of signals x-y, y-z and x-z, Mutual information is calculated between pairs of signals x-y, x-z, x-m, y-z, y-m and z-m, and Correlation coefficients are calculated between pairs of signals x-y, y-z and x-z.

- **Swipe Features:** For each swipe performed by users on tablet and phone, various features related to the speed and trajectory of the swipes are extracted. The last row of Table 4.2 summarizes them. The features include; the minimum and the maximum x and y coordinates; the Euclidean distance between the start and stop points; the tangent angle of the swipe; the total time taken to for the swipe; the mean and standard deviation and the quartiles of velocity, acceleration, pressure and area; and the direction of the swipe used to group them into horizontal or vertical swipes.

#### 4.4 Experimentation and analysis

In this section, we describe the tests and the procedure we use to examine the hypothesis that a sample comes from normal distribution.

**Non-parametric tests:** To test if the feature values have an underlying normal distribution, we use two non-parametric tests of normality namely; **a) Lilliefors test** [100], which is a modified form of Kolmogorov–Smirnov test [111] suitable for large datasets for non-parametric testing of the null hypothesis that the data comes from a normally distributed

population; and **b) Shapiro-Wilk test** [159] which is also a non-parametric test and is more suitable for testing the null hypothesis on smaller data ( $n < 50$ ).

**Hypothesis testing:** For all tests we begin with the null hypothesis  $H_0$ , that the sample came from a normal distribution.  $H_0$  can be discarded if the p-value from the test is below the critical value of 0.05.

**Sampling and Testing Procedure:** In our tests, we draw random samples consisting of 75% of the feature set for each modality and for each user. We then perform a suitable normality test (Lilliefors test if number of samples  $> 50$  or Shapiro-Wilk test otherwise) and store the p-values with corresponding annotations. We repeat the process ten times for all features, modalities and users to arrive at more accurate conclusions for evidence of normality in the underlying distribution. This form of random sampling is inspired from many other statistical research ([22, 93, 154]) that show that it is hard for goodness-of-fit tests to provide meaningful results on large datasets due to their loosely fitting statistical descriptions.

**Grouping of Features:** As the tests help us to discard  $H_0$ , based on the percentage of samples that had p-value  $> 0.05$  we categorize features into four broad categories; a) less than 25% of samples with p-value  $> 0.05$ ; b) 25% to 50% of samples with p-value  $> 0.05$ ; c) 50% to 75% of samples with p-value  $> 0.05$ ; and, d) above 75% samples with p-value  $> 0.05$ . The values have been color coded based on these categories for visual clarity.

## 4.5 Results and discussion

For a majority of the test samples for features in behavioral biometrics, we found that  $H_0$  could be discarded, implying the samples did not belong to a Gaussian distribution. In the case of keystroke features we found that all but one feature were in the first category with less than 25% samples where  $H_0$  could not be rejected. The detailed results for each feature and device are shown in Table 4.3 to Table 4.6. While Table 4.3 shows the percentage of tests with  $p > 0.5$  for unigraphs on desktop, tablet and phone, Table 4.4 to 4.6 show the results for digraphs. Figure 4.1a summarizes all the features for typing activity with respect to their categories. In table 4.3, we observe that the highest percentage of samples where  $H_0$  could not be rejected was for unigraph "n", with about 27%, which is still a very low number of tests when compared to the total number in our experiments, and thus can be ignored. As the case with desktop was an expected result following the lines of work in [154], these results reaffirm the findings and extend it further to hand-held devices such as phone and tablet.

In the case of swiping activity, we first separated the samples into vertical and horizontal swipes before testing the features for normality. We found a negligible amount of them belonged to the fourth category (above 75% samples where  $p > 0.05$ ) on phone. While most of the features belonged to the first and second category of; less than 25%; and, 25% to 50% of samples with  $p > 0.05$  respectively. Table 4.7, shows the results for individual features and Figure 4.1b summarizes the categorical distribution for both tablet and phone.

Samples from gait activity were further divided into walking (on flat corridor), downstairs and upstairs samples before testing each for normality. We found that upstairs and downstairs samples had considerable number of features in the third and fourth categories; 50% to 75%; and, above 75% samples where the  $H_0$  could not be rejected respectively. This was observed on all three devices used for these tasks: Hand-Phone, Pocket-Phone and Hand-Tablet. However, for the walking tasks majority of the features belonged to the first category and had less than 25% samples with  $p > 0.05$ . Individual values for the features of walking, downstairs and upstairs are shown, in Tables 4.10, 4.9 and 4.8 respectively. Their summary is illustrated in Figures 4.1d, 4.1e and 4.1c respectively. It is intriguing why data from upstairs and downstairs activities behaved so differently, we discuss this in the following section. From our results it is clear that features extracted from the behavioral biometrics data for typing, gait and swiping activities on desktop, tablet and phone do not follow a normal distribution. Elaborate discussion on the implications of our results and alternate approaches of non-normality follow.

#### **4.6 Conclusion and alternate approaches**

The implications of assuming a normal distribution in data when the data is actually from a different distribution have been studied across various domains ([1, 37, 101, 128, 181]). If methods wrongly assume a normal distribution the findings may be misleading or wrong. In the past, several studies in behavioral biometrics have assumed a normal distribution in data ([164, 167]) and could improve results by either extracting features that followed normal distribution or by implementing methods more suitable for non-normal distributions

[99]. Our experiments show that, in the case of keystrokes, gait and swipes using desktop, tablet and phone, it would be wrong to assume an underlying normal distribution. Low values of  $p$  from our non-parametric normality tests across activities and devices show that researchers in behavioral biometrics must not assume the data to be from a Gaussian distribution to get better and more accurate insights. However, upstairs and downstairs activity data, showing higher percentages of samples where an underlying normal distribution cannot be discarded is intriguing and further research is needed to establish why this occurs. Knowing that the data does not follow normal distribution leaves the discussion incomplete, which can only be completed by learning alternate ways to handle a non-normal dataset.

**Alternate approaches and solutions:** One should first test for conditions of normality in data before making such an assumption. If the conditions are not met, there are numerous ways to work around the absence of normal distribution. We discuss concepts that have been successfully applied in related fields and other intuitive methods that researchers can use to perform more accurate analysis on behavioral biometric data. With respect to analysis and modelling of the data itself, there are several techniques found in literature. Different types of distributions like Weibull, Gamma, Exponential or Pareto distributions have been used, with theoretical and empirical justifications ([2, 105]), however, a goodness of fit test beforehand is advised. Methods like Heteroscedastic Corrected Covariance Matrix, Bootstrapping are forms of altering the estimator to better describe non-normal data. Winsorizing and trimming of data is an intermediate technique that replace the parameters of the original data by virtue of replacing extremities in a sample. Data transformation methods like Johnson Transformation [192], Box-Cox Transformations [191] and

other forms of algorithmic and parametric transformations [137] have shown good results in other data intensive researches.

Apart from data modelling and transformation, in research involving identification, verification or classification tasks, attention to the choice of classifiers can greatly improve performance. It is common for researchers to use Gaussian Naive Bayes classifiers or Linear classifiers with Linear Discriminant Analysis, which are very popular tools for baseline metrics. However, these classifiers assume that the data has an underlying normal distribution and lack of such a distribution can cause their performance to deteriorate heavily. Modified, non-parametric versions of Gaussian Naive Bayes classifier described in [165], have shown to perform well. Standard classifiers, that are not designed with the assumption of normality in data, like Support Vector Machines, K Nearest Neighbor Classifiers or Neural Networks are intuitively a better choice for researchers in behavioral biometrics.

Our results question the common assumption that the data in behavioral biometrics follows a normal distribution. We have discussed the implications and alternate approaches for such a scenario. We hope that, insights from our work help future researchers to make the right choices in terms of data models, transformations and classifiers to achieve better results and make correct interpretations. We come full circle to the question that we began with, "Should the *assumption of normality* be the *norm* in behavioral biometrics?", and equipped with the knowledge from our experiments discussed above, we answer, "no", with a caveat that careful examination of the validity of assumptions about underlying distributions is a must.

To maintain clarity in presentation we use symbols  $d1$  through  $d18$  to denote the digraphs ( $d1$ : ('BACKSPACE', 'BACKSPACE'),  $d2$ : ('SPACE', 'a'),  $d3$ : ('SPACE', 'i'),  $d4$ : ('SPACE', 's'),  $d5$ : ('SPACE', 't'),  $d6$ : ('e', 'SPACE'),  $d7$ : ('e', 'n'),  $d8$ : ('e', 'r'),  $d9$ : ('e', 's'),  $d10$ : ('n', 'SPACE'),  $d11$ : ('o', 'SPACE'),  $d12$ : ('o', 'n'),  $d13$ : ('r', 'e'),  $d14$ : ('s', 'SPACE'),  $d15$ : ('s', 'e'),  $d16$ : ('t', 'SPACE'),  $d17$ : ('t', 'e') and  $d18$ : ('t', 'h'))

Table 4.3: Percentage of test samples with  $p > 0.05$  for keyhold feature from unigraphs on desktop, tablet and phone.

Unigraph	Desktop	Tablet	Phone
SPACE	0%	1%	11%
BACKSPACE	0%	0%	2%
a	0%	15%	20%
e	0%	0%	5%
h	0%	9%	18%
i	0%	2%	9%
l	0%	9%	15%
n	0%	2%	27%
o	0%	5%	22%
r	0%	7%	22%
s	0%	4%	16%
t	0%	1%	9%

Table 4.4: Percentage of test samples with  $p > 0.05$  for flight1-flight4 features from digraphs on desktop.

Digraph	Desktop			
	Flight1	Flight2	Flight3	Flight4
$d1$	0%	0%	0%	0%
$d2$	0%	0%	0%	0%
$d3$	1%	2%	1%	1%
$d4$	0%	0%	0%	0%
$d5$	0%	0%	0%	0%
$d6$	0%	0%	0%	0%
$d7$	2%	3%	3%	2%
$d8$	0%	2%	2%	3%
$d9$	0%	1%	3%	4%
$d10$	0%	0%	1%	0%
$d11$	1%	1%	1%	1%
$d12$	3%	2%	2%	4%
$d13$	0%	2%	1%	3%
$d14$	0%	0%	0%	1%
$d15$	0%	1%	0%	0%
$d16$	0%	0%	0%	0%
$d17$	2%	5%	4%	6%
$d18$	0%	2%	0%	0%

Table 4.5: Percentage of test samples with  $p > 0.05$  for flight1-flight4 features from digraphs on tablet.

Digraph	Tablet			
	Flight1	Flight2	Flight3	Flight4
d1	0%	0%	0%	0%
d2	0%	0%	0%	0%
d3	2%	1%	1%	1%
d4	1%	1%	2%	1%
d5	0%	0%	0%	0%
d6	1%	1%	1%	1%
d7	5%	7%	6%	9%
d8	9%	7%	13%	11%
d9	5%	3%	4%	4%
d10	3%	1%	2%	3%
d11	0%	1%	0%	1%
d12	8%	6%	3%	9%
d13	5%	4%	5%	5%
d14	0%	0%	0%	1%
d15	4%	3%	3%	4%
d16	0%	0%	0%	0%
d17	18%	17%	16%	19%
d18	8%	6%	3%	3%

Table 4.6: Percentage of test samples with  $p > 0.05$  for flight1-flight4 features from digraphs on phone.

Digraph	Phone			
	Flight1	Flight2	Flight3	Flight4
d1	0%	0%	0%	0%
d2	1%	0%	2%	0%
d3	0%	0%	1%	1%
d4	1%	1%	0%	0%
d5	0%	0%	0%	0%
d6	0%	1%	1%	1%
d7	6%	5%	8%	8%
d8	15%	17%	22%	23%
d9	5%	8%	6%	11%
d10	0%	2%	1%	1%
d11	1%	3%	2%	4%
d12	14%	16%	15%	17%
d13	7%	5%	9%	12%
d14	0%	3%	1%	3%
d15	0%	6%	2%	8%
d16	1%	1%	1%	1%
d17	18%	20%	19%	17%
d18	6%	4%	6%	6%

Table 4.7: Percentage of test samples with  $p > 0.05$  for features from Swiping activity on phone and tablet.

Features	Phone	Tablet	Features	Phone	Tablet
minx	2%	3%	aquarts_0	1%	7%
miny	21%	32%	aquarts_1	0%	0%
maxx	5%	2%	aquarts_2	0%	0%
maxy	6%	16%	pmean	72%	26%
eucliddist	40%	26%	pstd	76%	33%
tanangle	0%	1%	pquarts_0	65%	29%
tottime	0%	0%	pquarts_1	65%	26%
vmean	2%	9%	pquarts_2	66%	21%
vstd	6%	14%	areamean	56%	33%
vquarts_0	0%	1%	areastd	46%	37%
vquarts_1	0%	5%	areaquarts_0	9%	1%
vquarts_2	2%	14%	areaquarts_1	4%	0%
amean	0%	0%	areaquarts_2	4%	0%
astd	0%	0%			



Table 4.8: Percentage of test samples with  $p > 0.05$  for features from Upstairs activity with phone in hand, phone in pocket and tablet in hand.

Feature	Hand_Phone		Pocket_Phone		Hand_Tablet	
	Acc	Gyr	Acc	Gyr	Acc	Gyr
xmean	79%	76%	62%	67%	74%	78%
ymean	83%	60%	77%	20%	79%	56%
zmean	89%	16%	56%	51%	80%	11%
mmean	89%	63%	75%	60%	91%	44%
xstd	68%	79%	67%	60%	72%	83%
ystd	75%	76%	69%	64%	72%	79%
zstd	79%	85%	66%	62%	88%	85%
mstd	85%	63%	65%	57%	91%	61%
xbp	67%	72%	67%	62%	61%	68%
ybp	74%	49%	77%	45%	64%	59%
zbp	85%	21%	69%	56%	80%	19%
mbp	85%	44%	74%	56%	86%	28%
xenergy	68%	71%	68%	62%	62%	67%
yenergy	74%	49%	76%	45%	60%	59%
zenergy	85%	21%	69%	56%	76%	19%
menergy	85%	46%	76%	56%	71%	28%
xmfreq	31%	11%	61%	44%	4%	19%
ymfreq	25%	16%	63%	43%	16%	11%
zmfreq	36%	16%	52%	26%	3%	5%
mmfreq	86%	81%	76%	79%	96%	91%
xizr	74%	80%	74%	60%	74%	86%
yizr	71%	79%	74%	74%	70%	79%
zizr	79%	83%	76%	69%	85%	83%
miqr	87%	67%	70%	70%	85%	52%
xrange	70%	68%	50%	58%	65%	74%
yrange	68%	70%	51%	62%	69%	75%
zrange	73%	80%	60%	56%	72%	73%
mrange	72%	60%	53%	53%	70%	71%
xsnr	78%	74%	61%	68%	74%	80%
ysnr	81%	62%	44%	27%	85%	59%
zsnr	72%	19%	63%	66%	63%	12%
msnr	71%	78%	53%	74%	59%	85%
xydtw	69%	79%	74%	73%	62%	77%
yzdtw	86%	46%	71%	55%	78%	38%
xzdtw	81%	30%	74%	69%	76%	20%
xymi	86%	88%	74%	78%	90%	86%
xzmi	89%	85%	79%	77%	87%	85%
xmmi	87%	80%	79%	74%	87%	90%
yzmi	87%	90%	79%	74%	88%	85%
ymmi	89%	84%	67%	75%	91%	89%
zmmi	88%	83%	76%	79%	81%	82%
xycorr	88%	92%	67%	61%	86%	91%
yzcorr	85%	91%	63%	56%	87%	85%
xzcorr	91%	85%	69%	46%	88%	91%

Table 4.9: Percentage of test samples with  $p > 0.05$  for features from Downstairs activity phone in hand, phone in pocket and tablet in hand.

Feature	Hand_Phone		Pocket_Phone		Hand_Tablet	
	Acc	Gyr	Acc	Gyr	Acc	Gyr
xmean	84%	85%	74%	72%	74%	85%
ymean	85%	61%	81%	14%	83%	61%
zmean	88%	24%	57%	76%	84%	21%
mmean	90%	78%	90%	72%	91%	56%
xstd	81%	87%	79%	80%	74%	87%
ystd	86%	74%	83%	75%	78%	87%
zstd	89%	85%	82%	76%	89%	83%
mstd	91%	83%	80%	74%	87%	68%
xbp	77%	72%	80%	77%	60%	79%
ybp	82%	59%	85%	50%	67%	70%
zbp	86%	32%	85%	76%	81%	14%
mbp	91%	62%	90%	67%	89%	48%
xenergy	77%	71%	80%	75%	60%	77%
yenergy	81%	61%	87%	48%	68%	70%
zenergy	87%	31%	85%	76%	81%	15%
menergy	89%	62%	88%	68%	88%	44%
xmfreq	24%	20%	68%	50%	6%	8%
ymfreq	27%	21%	68%	40%	16%	19%
zmfreq	48%	15%	72%	33%	1%	12%
mmfreq	91%	91%	83%	88%	98%	92%
xiqr	86%	91%	83%	82%	70%	90%
yiqr	82%	85%	84%	79%	73%	89%
ziqr	85%	85%	85%	84%	89%	78%
miqr	90%	74%	82%	78%	88%	59%
xrange	79%	74%	67%	64%	70%	74%
yrange	77%	65%	64%	71%	70%	79%
zrange	72%	82%	69%	70%	76%	79%
mrangle	77%	72%	71%	56%	74%	76%
xsnr	90%	76%	72%	72%	79%	72%
ysnr	78%	57%	50%	19%	81%	61%
zsnr	56%	29%	74%	77%	56%	32%
msnr	53%	84%	54%	74%	55%	79%
xydtw	81%	76%	85%	73%	71%	87%
yzdtw	84%	73%	79%	67%	84%	44%
xzdtw	85%	60%	82%	78%	78%	27%
xyymi	85%	84%	79%	78%	85%	85%
xzmi	89%	85%	80%	76%	85%	80%
xmmi	90%	83%	80%	74%	91%	75%
yzmi	83%	85%	80%	75%	78%	85%
ymmi	87%	89%	75%	80%	88%	88%
zmmi	89%	84%	79%	78%	71%	84%
xycorr	86%	93%	78%	79%	88%	91%
yzcorr	87%	93%	77%	79%	82%	89%
xzcorr	91%	85%	84%	76%	89%	92%

Table 4.10: Percentage of test samples with  $p > 0.05$  for features from Walking activity phone in hand, phone in pocket and tablet in hand.

Feature	Hand_Phone		Pocket_Phone		Hand_Tablet	
	Acc	Gyr	Acc	Gyr	Acc	Gyr
xmean	36%	25%	21%	6%	32%	28%
ymean	39%	9%	28%	6%	38%	11%
zmean	32%	4%	16%	12%	25%	2%
mmean	33%	18%	28%	8%	34%	10%
xstd	35%	41%	9%	3%	37%	44%
ystd	21%	15%	9%	13%	31%	32%
zstd	37%	25%	8%	7%	38%	19%
mstd	38%	16%	5%	9%	40%	4%
xbp	3%	9%	35%	11%	4%	12%
ybp	15%	2%	31%	15%	11%	2%
zbp	37%	3%	27%	26%	25%	0%
mbp	31%	5%	24%	13%	36%	3%
xenergy	7%	18%	15%	3%	11%	27%
yenergy	26%	3%	3%	12%	23%	4%
zenergy	0%	2%	15%	21%	0%	0%
menergy	0%	8%	0%	2%	0%	3%
xmfreq	3%	0%	14%	8%	0%	0%
ymfreq	1%	0%	17%	6%	0%	0%
zmfreq	6%	1%	11%	3%	0%	0%
mmfreq	65%	58%	62%	68%	98%	74%
xiqr	26%	38%	26%	10%	35%	52%
yiqr	29%	16%	14%	22%	38%	37%
ziqr	50%	24%	30%	17%	50%	27%
miqr	53%	11%	12%	15%	54%	4%
xrange	29%	36%	9%	3%	36%	38%
yrange	26%	17%	10%	9%	26%	28%
zrange	30%	23%	9%	11%	27%	25%
mrange	32%	20%	7%	4%	23%	13%
xsnr	17%	10%	5%	1%	15%	12%
ysnr	12%	13%	3%	3%	11%	3%
zsnr	7%	8%	3%	4%	5%	12%
msnr	7%	31%	1%	9%	3%	32%
xydtw	17%	24%	3%	15%	17%	28%
yzdtw	18%	16%	3%	8%	4%	15%
xzdtw	3%	5%	9%	13%	2%	2%
xymi	0%	12%	0%	1%	9%	23%
xzmi	0%	5%	2%	1%	3%	17%
xmmi	0%	0%	2%	2%	0%	2%
yzmi	0%	9%	2%	1%	3%	11%
ymmi	0%	0%	0%	2%	1%	1%
zmmi	0%	0%	2%	1%	0%	2%
xycorr	55%	50%	19%	8%	65%	60%
yzcorr	45%	56%	12%	11%	59%	53%
xzcorr	46%	16%	12%	9%	57%	42%

## **5. CLASSIFICATION OF THREAT LEVEL IN TYPING ACTIVITY THROUGH KEYSTROKE DYNAMICS**

System intrusion is a major issue in today's data-driven world. Discovering adversarial activities before or as they happen, through any modality, makes a system more secure. A straight forward approach to determine threat level from typing data would be to analyze the text directly. However, natural language processing is hard to implement on complex real life data, therefore we explore how far keystroke dynamics can go in terms of classifying threat levels correctly.

User interactions with modern day systems occur through various modalities. Keyboards are the most popular choice of user input on desktops. The typing behavior of a user also called Keystroke Dynamics (KD) has been of great interest among researchers in the recent past. Studies have shown that KD can be used as a means of user authentication [13, 175]. KD has gained the attention of researchers as a popular behavioral-biometric due to the popularity and ease of use of keyboards. One can see the contribution of keystroke dynamics in diverse fields such as continuous authentication, user identification, and verification[5, 13, 50, 78, 125, 147, 149]. A user can accomplish various tasks through keystroke inputs. Intuitively some activities are benign in nature while others are malicious. Common day-to-day activities like writing emails, documents or browsing the internet can possess a lesser threat to the system from the user, whereas activities involving terminal commands may possess greater threats.

System intrusion detection has been explored by many researchers [31, 150] who have proposed solutions at various levels of system interaction, ranging from system calls to data mining techniques [3, 189]. But the possibility of using behavioral biometrics, in general, or typing behavior, in particular, to detect malicious activity has not been explored.

We show that a user's typing behavior can reveal important information to help secure a system. We classifying the nature of the typing activity into two broad categories; benign or adversarial. The shortcomings of conventional keystroke features for this task are analyzed and 14 features that are more suitable for this classification are proposed and evaluated. Results are presented from experiments on the typing data from over a hundred users (in each activity category) performing benign and malicious tasks in separate sessions.

## 5.1 Key contributions of the chapter

Our key contributions are;

- ***Classify type of keystroke activity:*** We show that a user's typing behavior can be used to classify the origin of a text sample as benign or adversarial. We use a mixture of keystroke timings and content related features to achieve good classification performance.
- ***Propose and evaluate new features for threat level classification:*** Results from conventional features show that they are inefficient for the problem we attempt to solve. Therefore, we propose 14 new features designed specifically for our prob-

lem and evaluate them with three different classifiers and four different text sample sizes. We find our proposed feature to work well in all cases.

## 5.2 Related work

Use of KD has been explored in various applications [45, 56, 175]. It has been effective in the identification and verification of users [14, 78]. A list of common key strings is used for identification of each user. The authors found that non-English sequence of characters was more accurate than the English sequence. Karnan and Krishnaraj [85], discuss using keystroke dynamics in mobile authentication by analyzing habitual rhythm patterns in the way users type. Keystroke Dynamics was also shown to be effective in continuous authentication and well adapted for desktop devices [39].

A study on keystroke biometric Identification on Long-Text [175], shows that a user identification was highly accurate in long text inputs (copy and free text) with the same keyword used for both enrolment and testing. There was an insignificant decrease in the accuracy of free text when compared to copied text. In a study on free texts based on features such as left-hand keys, right-hand keys, flight time, and percentage of special key characters, they discovered that the accuracy of copy task was higher compared to free text typing. This also purports that the type of text and the situation affects the user typing behavior[13].

Gunetti and Picardi[69] remark that one can analyze the typing rhythms of an individual in free text. In another work on anomaly intrusion detection based on biometrics, an intrusion detection system was built using keystroke biometrics and mouse dynamics. Dwell

Table 5.1: Highlights of our data collection effort.

	<b>Dataset 1: Benign activity</b>	<b>Dataset 2: Adversarial activity</b>
<b>Number of Users</b>	102	103
<b>Tasks Performed</b>	<ul style="list-style-type: none"> <li>a) Transcription (fixed text)</li> <li>b) Browsing (shopping for a list of items)</li> <li>c) Note taking (item prices, site links, etc.)</li> <li>d) Short answers to a list of questions (free text)</li> </ul>	<ul style="list-style-type: none"> <li>a) Browsing (To help with adversarial tasks)</li> <li>b) Network Discovery</li> <li>c) Target Identification</li> <li>d) Password Dictionary Attack</li> <li>e) Privilege Escalation</li> <li>f) Data Exfiltration</li> </ul>
<b>Input/Output</b>	Standard QWERTY Keyboard (Dell kb212-b) Optical Mouse (Dell ms111-p) Dell 21 inch Monitor	Standard QWERTY Keyboard (Dell kb212-b) Optical Mouse (Dell ms111-p) Dell 21 inch Monitor x 2
<b>Operating System</b>	Windows 7	Kali Linux
<b>Average keystrokes per user</b>	$\simeq 12,210$	$\simeq 6,866$
<b>Average duration of each user's session</b>	$\simeq 55$ minutes	$\simeq 1\text{Hr } 30$ minutes
<b>Demographics</b>	Gender: Male = 65 ; Female = 37 Age: 19 - 35 years (Avg. = 24.9 and StD. = 3.1) Typing Style: Touch = 30 ; Visual = 72	Gender: Male = 82 ; Female = 21 Age: 18 - 32 years (Avg. = 24.2 and StD. = 3.3) Typing Style: Touch = 37 ; Visual = 66
<b>Span of data collection</b>	$\simeq 3$ Months	$\simeq 4$ Months

time and flight time were chosen as their feature sets. These feature sets were later rendered into digraphs and trigraphs for interpretation. The legitimate users were validated by the algorithm successfully [5].

Researchers have been exploring new methods to address the issue of system intrusion. A work by Yampolskiy[189] employs indirect human-computer interaction based biometrics such as audit logs, call stack data, network traffic, system calls, GUI interaction, registry access data, storage activity in their work. They have explained how network traffic data can be analyzed to build an intrusion detection system[189]. In another work, authors use two different mining methods, an apriori algorithm and sequence mining, for intrusion detection [3].

Zamonsky Pendernera et al.[194] used keystroke dynamics for intruder classification. The detection of intruders was done using clustering methods on the keystroke data, which falls into the umbrella of verification. Our work, in contrast, aims at classifying if the activity performed is itself malicious in nature irrespective of the user. Another study [184] on Detecting intrusions using system calls examined diverse approaches such as Enumerating Sequences, Frequency-based methods, Data mining approaches and Finite State Machines for distinguishing normal behavior from the intrusions utilizing system calls as a feature. The authors opined that the choice of the data stream is critical. Though the intent of the authors in this work is similar to ours, detection of malicious activity through behavioral biometrics has not been explored.

Our work stands apart from other works explored in our literature survey in that we focus on the typing behavior of a user to classify if the typing activity is benign or adversarial.



### **5.3 Dataset and experimentation methods**

In this section we elaborate our data collection, experiments with conventional features and our proposed features. The collected data will be made publicly available soon for the benefit of the research community.

#### **5.3.1 Details of the data collection**

Two separate data collection efforts were carried out at our university after the IRB approval. The focus of both efforts was to capture keystroke data from two different categories of activity that a user can indulge in on a desktop. The first collection effort involved benign activities that most users carry out on any regular day, whereas the second collection effort involved adversarial activities generally performed with malicious intent. In both cases, the participants were allotted unique user IDs for their sessions and were asked to fill out a brief demographic survey form. A summary of the data collection and key statistics are described in Table 5.1.

The first category of keystroke data from benign-activity was collected from 102 users who performed various day to day tasks on a desktop such as; a) Transcription of fixed text, b) Shopping online for a list of items, c) Note taking and d) Writing short free-text answers to a list of questions with varying cognitive loads. The desktop had standard input and output components such as a full-size QWERTY keyboard, an optical mouse and a 21-inch monitor. On an average, each user performed about 12,210 keystroke events (key-press and key-release) in their session which lasted about 55 minutes. All keystroke and mouse events were logged using simple windows interrupt hooks. The participant

population consisted of 65 males and 37 females, of which 30 and 72 participants identified themselves as touch typists and visual typists respectively. The participants were from the age group of 19 to 35 years and the data collection was completed over a span of 3 months. This is a subset of SU-AIS BBMAS dataset [83].

The second data collection effort to capture keystroke data from adversarial activity had 103 users. 15 users from the first data collection also participated in the second data collection. However, we do not separate their data and process them without any special identifiers. The users performed various adversarial activities aided by some assistance in the form of web pages that provided directions and hints when needed. A virtual environment was setup to mimic virtual systems on a network. The users had to perform attacks by acquiring credentials to systems in the network logging into them and stealing any important files from the compromised systems. The key tasks involved were; a) Browsing: the users were allowed to browse the internet for any information to help complete the tasks in the session, b) Network discovery: identifying systems and services on the network, c) Target identification: identifying a target with vulnerabilities, d) Password dictionary attack: carrying out a password dictionary attack to find the username-password combination and logging on to the target. e) Privilege escalation: escalating privileges on the target system to root and f) Data exfiltration : extract any personal data from the victim on to the host. The input and output components were similar to the first data collection except for the addition of a second monitor. The users were instructed to browse read instructions on one monitor and carry out the attack on the secondary monitor. On an average, each user made about 6,866 keystrokes in their session which last approximately for an hour and a half. 82 participants were male and 21 were female. 37 partici-

pants identified themselves to be touch typists while the rest marked themselves as visual typists. This entire data collection effort spanned for about 4 months.

For the scope of this work, the entire dataset collected from benign activities (hereinafter referred to as benign-dataset) is considered as one class of data, whereas dataset collected from adversarial activities (hereinafter referred to as adversarial-dataset) is considered to be the other class. Text samples of varying lengths drawn from these datasets are called benign-sample and adversarial-sample respectively. All together, our datasets consist of more than 1.9 million keystrokes from more than 200 individual sessions making it one of the richest datasets on keystrokes.

***Outlier Removal:*** For the detection and removal of outliers, we use a simple filter to remove any instances of keys that were held down for two seconds or more. We also remove instances of the inter-key pauses that are greater than two seconds. We assume that these were caused by pauses, where the user is either thinking or receiving instructions during the data collection.

### 5.3.2 Context recognition with conventional features

Figure 2.4 illustrates the temporal features. We select the ten most commonly occurring unigraphs and five digraphs in all text samples from our dataset. The unigraphs are : 'a' , 't' , 'h' , 's' , 'space' , 'i' , 'n' , 'r' , 'e' and 'l' and the digraphs are: 'e,r' , 's,s' , 't,h' , 's,space' , 'space,a' and extract their conventional features. We also use four different sizes for the text samples extracted from both the datasets. The text samples sizes are varied from small text samples of 100 characters (200 keystrokes) to large text samples of 1000 char-

acters (2000 keystrokes). From each of the samples, the shortlisted unigraphs and digraphs are considered, the *Keyhold* values are extracted from unigraphs and the *Flight* values are extracted from the digraphs. For each of the unigraphs and digraphs, their mean, standard deviation and median of the corresponding values are used to form a feature vector that represents the text sample. The feature vector has a total of 45 columns ([10 unigraphs + 5 digraphs] \* [mean, standard deviation, median]) excluding the class and sample length labels. The origin of the text sample (benign or adversarial) is then marked as its class label. After extracting all the text samples for text lengths: 100, 250, 500 and 1000, forming the feature vectors for each of them and assigning their respective class labels, we carry out simple two-class classification experiments. The experiments for different text sample sizes are conducted separately. The classifiers were trained to detect adversarial samples, therefore classification of a benign sample as adversarial is a False Positive and the opposite is considered False Negative.

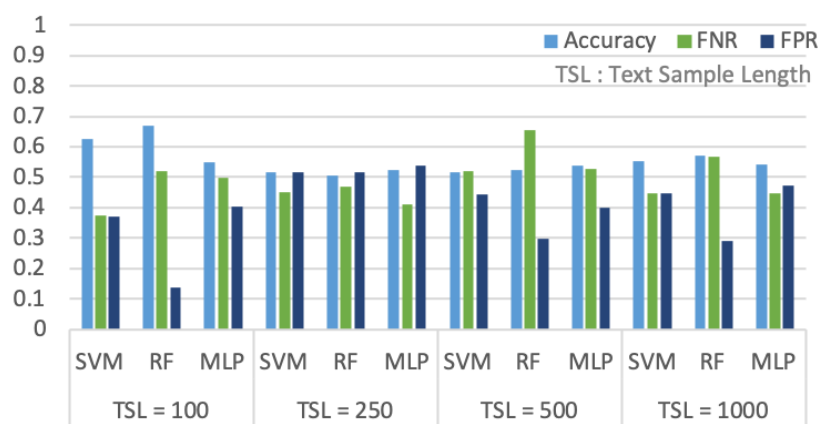


Fig. 5.1.: The accuracies, FPRs (Type 1 error) and FNRs (Type 2 error) from the SVM, RF and MLP classifiers trained and tested using the conventional keystroke features.

Using 70% of samples for training and rest 30% of them for testing, we train 3 different classifiers; Support Vector Machine (SVM), Random Forest (RF) and Multilayer Perceptron (MLP). SVM classifier was set with a linear kernel and a penalty parameter  $C=1$ . The RF classifier was restricted to 5 decision trees with maximum depths of 5 and a maximum of 2 leaves per node. The split criterion was set to Gini Impurity and only one feature was used for the split at each node. The MLP classifier had a single hidden layer with 10 nodes and 'relu' activation function with 'adam' optimizer was used. The batch size was set to 200 and the MLP was trained for 1000 iterations with a learning rate of 0.001. Using SMOTE [40] to balance the test and training sets, we use approximately 3000 samples training and 900 samples for testing from each class in case of sample length equal to 100 characters. In the case of sample length equal to 200, we use about 1230 samples for training and 370 samples for testing from each class. For sample lengths of 500, we use 550 samples to train and 165 samples to test from each class. For samples with length equal to 1000 we use about 300 samples to train and 90 to test from each class.

**Observations:** With the conventional features we observe very low accuracies in classification and high Type 1 errors (False Positives: when a text sample is classified as adversarial but it is actually benign) and Type 2 errors (False Negatives: when a text sample is classified as benign but it is actually adversarial). Figure 5.1 shows the performance measures for various lengths of text samples. In all cases, the accuracies were close to random (50%) and in most cases, the errors were high in the range of 35% to 50%. It is clear that features that have performed well for user identification and verification are not suitable for the task of classifying keystroke behavior into benign or adversarial.

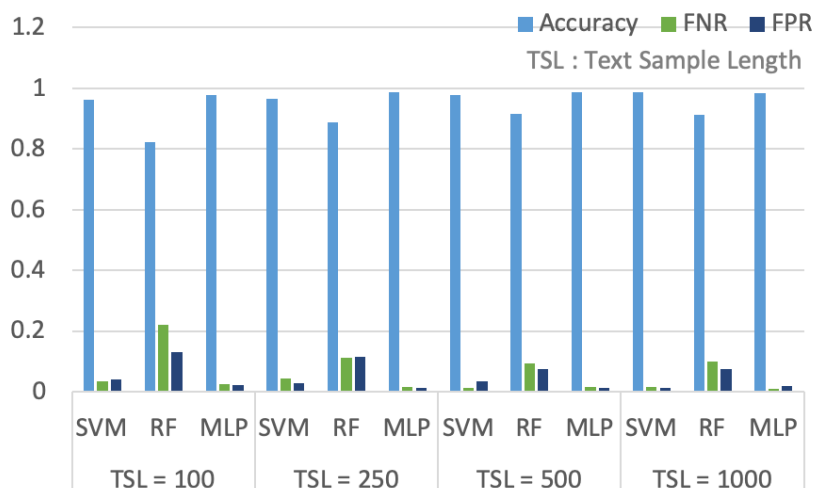


Fig. 5.2.: The accuracies, FPRs (Type 1 error) and FNRs (Type 2 error) from the SVM, RF and MLP classifiers trained and tested using our proposed keystroke features.

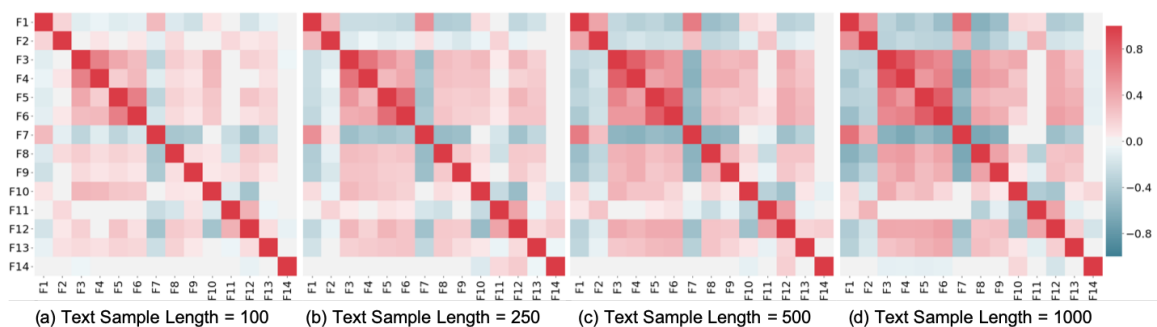


Fig. 5.3.: Heat-maps showing the correlation between feature pairs in the proposed feature set for different sizes of text samples.  $F1$  to  $F14$  on the x-axis and y-axis represent the features *AvgEnterHold*, *StdEnterHold*, *AvgSpaceInFlight*, *StdSpaceInFlight*, *AvgSpaceOutFlight*, *StdSpaceOutFlight*, *SpaceRatio*, *EnterRatio*, *ErrorCount*, *TotalTime*, *IQRHold*, *IQRFlight*, *PunctuationRatio* and *SpeedDelta*, respectively

### 5.3.3 Proposed features

We propose a set of 14 features that were heuristically evaluated on our datasets for classifying text samples into benign or adversarial samples. The features are a mixture of temporal and content based features and reflect the change in typing behavior and content being typed between benign and adversarial activity.

Our proposed features are described below, followed by their category (temporal or content based):

- a. *AvgEnterHold*: The average of *Keyhold* values of all *Enter* keys in the text sample. (Temporal)
- b. *StdEnterHold*: The standard deviation of *Keyhold* values of all *Enter* keys in the text sample. (Temporal)
- c. *AvgSpaceInFlight*: The average of *Flight1* values of any key followed by *space*. (Temporal)
- d. *StdSpaceInFlight*: The standard deviation of *Flight1* values of any key followed by *space*. (Temporal)
- e. *AvgSpaceOutFlight*: The average of *Flight1* values when *space* precedes any key. (Temporal)
- f. *StdSpaceOutFlight*: The standard deviation of *Flight1* values when *space* precedes any key. (Temporal)
- g. *SpaceRatio*: The number of times *space* occurs / length of the text sample. (Content)
- h. *EnterRatio*: The number of times *Enter* occurs / length of the text sample. (Content)
- i. *ErrorCount*: The number of times *Backspace* or *Delete* occurs in the text sample. (Content)

- j. *TotalTime*: Sum of all *Keyhold* and *Flight1* values in the text sample. (Temporal)
- k. *IQRHold*: The Inter-Quartile-Range of all the *Keyhold* values in the text sample. (Temporal)
- l. *IQRFlight*: The Inter-Quartile-Range of all the *Flight1* values in the text sample. (Temporal)
- m. *PunctuationRatio*: The number times punctuation occur / length of the text sample. (Content)
- n. *SpeedDelta*: The total time taken for the second half of the text sample - The total time taken for first half of the text sample. (Temporal)

#### **5.3.4 Context recognition with proposed features**

The classification experiments were rerun using our proposed features. The main goal of these experiments was to correctly classify the origin of a text sample either benign activity (Dataset 1) or adversarial activity (Dataset 2). A benign sample being classified as an adversarial sample is considered a False Positive (Type 1 error) and the opposite misclassification is a False Negative (Type 2 error).

For each size of the text sample, three different classifiers were trained using 70% of the text samples of the respective size (100, 250, 500 or 1000). The parameters of all three classifiers and the number of samples for training and testing are set to the same values as in previous experiments described in Section 5.3.2.



**Observations:** With our proposed features we observe high accuracies in classification and very low Type 1 errors (False Positives: when a text sample is classified as adversarial but it is actually benign) and Type 2 errors (False Negatives: when a text sample is classified as benign but it is actually adversarial). Figure 5.2 shows the performance measures for various lengths of text samples. In all cases the accuracies were very high with most values lying between 85% to 97%. We also observe that the RF classifier performed the poorest among the three classifiers, but even then had better results than the conventional features. The increase in the size of the text sample seems to help decrease the Type 1 and Type 2 errors in classification. In most cases, the errors were less than 15%. It is clear that our proposed features were suitable for detecting if a particular keystroke behavior originated out of benign or adversarial activity.

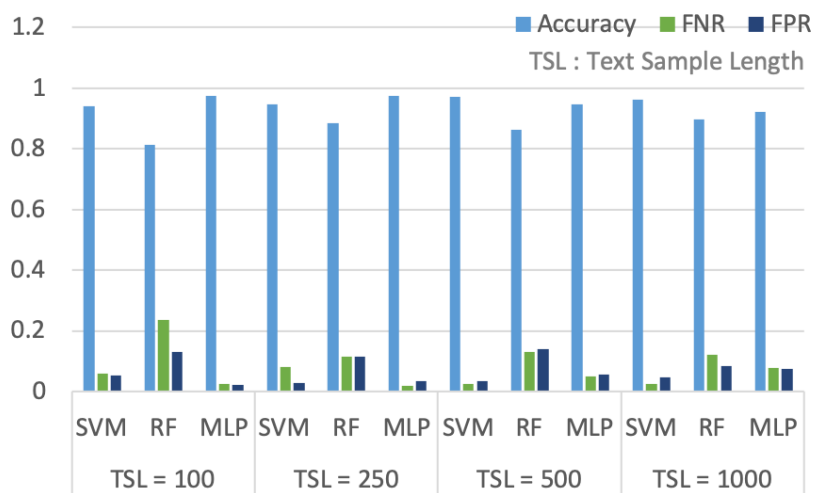


Fig. 5.4.: The accuracies, FPRs (Type 1 error) and FNRs (Type 2 error) from the SVM, RF and MLP classifiers using eight least correlated features from our proposed set.

### 5.3.5 Correlation analysis of feature pairs

To reduce our proposed feature set and analyze its trade-off we perform pairwise correlation of the proposed features. We use simple Pearson correlation value to depict how two features are linearly related. The pair-wise correlation analysis of features is done separately for each different size of the text sample. Figure 5.3 shows the heat-maps generated from our analysis. We retain the eight least correlated features which are *AvgEnterHold*, *AvgSpaceInFlight*, *AvgSpaceOutFlight*, *SpaceRatio*, *EnterRatio*, *ErrorCount*, *IQRFlight* and *PunctuationRatio*. The trade-off of eliminating the other six features is analyzed by training and testing the classifiers using only the eight selected features.

Figure 5.4 shows the accuracies and Type 1 and Type 2 error rates from the classifiers that used only the eight least correlated features for training and testing. All other parameters of the classifiers were unchanged. We observe that the classifiers performed very well and the reduction of six features does not seem to affect its performance drastically. Type 1 and Type 2 errors increase by a marginal amount which can be overlooked. On an average, the Type 1 errors increased in the range of 4% to 7% and Type 2 errors increased in the range of 3% to 6%. As a result the accuracies decreased but not to a noticeable extent. This clearly shows that even with a smaller set of eight features the text samples can be classified accurately to their activity of origin.

## 5.4 Conclusion and future work

We raised an intriguing question: "*Can the typing behavior of a user reveal if the typing activity is malicious or benign?*". We conclude that the typing behavior of a user can

reveal if the typing activity being done is benign or malicious. Although, the keystroke features that have been popularly used for user identification or verification are not suitable for this task. We proposed a different set of features using which the origin of a text sample (whether malicious or benign) could be determined with high levels of accuracies. We observe that behavior of keystroke timings and frequencies of certain keys like *Space*, *Enter* and *Punctuation* keys can be used to reveal the nature of typing activity. Using our proposed features we could achieve accuracies as high as 97% and Type 1 and Type 2 error rates of less than 3%. Our findings are based on the data that we collected from 102 users performing benign activity and 103 users performing malicious activity respectively. In total, our dataset has over 1.9 million keystroke events making it most suitable for such a study.

We show that keystroke analysis can be used to determine the nature of typing activity, thereby assessing the threat levels of a system. However, we understand that keystroke analysis would have to be used in conjunction with other technologies to obtain a more robust and secure system. A separate analysis on the 15 users who were common for both the data collection efforts might lead to interesting insights, we are exploring this direction. The future course of our research work would be to explore other input modalities which can reveal similar threats and to formulate a fusion of such modalities for a more secure system.

## 6. EXPLORATORY WORK

In this chapter we provide the details of our exploratory work carried out in two directions; a) Authentication of users through musical notes generated by mapping keystroke latencies to music and b) Authentication of users using the relationship between their keystroke latencies on different devices.

### 6.1 Authentication by Mapping Keystrokes to Music: The Melody of Typing

Expressing Keystroke Dynamics (KD) in form of sound opens new avenues to apply sound analysis techniques on KD. However this mapping is not straight-forward as varied feature space, differences in magnitudes of features and human interpretability of the music bring in complexities. We present a musical interface to KD by mapping keystroke features to music features. Music elements like melody, harmony, rhythm, pitch and tempo are varied with respect to the magnitude of their corresponding keystroke features. A pitch embedding technique makes the music discernible among users. Using the data from 30 users, who typed fixed strings multiple times on a desktop, shows that these auditory signals are distinguishable between users by both standard classifiers (*SVM*, *Random Forests* and *Naive Bayes*) and humans alike.

Behavioral biometrics has seen an upsurge in research and applications in the recent past. Behavioral biometrics such as keystrokes [20, 121, 158], touch and swipe [59, 116, 157],

gait patterns [64, 65, 183] and wrist movement patterns [97, 190] have been shown to be good second-factor authentication techniques. As humans have well developed auditory sense, representation of visual information in sound and vice versa has been of great interest to researchers [89, 112, 113, 144]. But such alternative interfaces to convey biometric information have not been explored. Mapping of information to sound can lead to deeper interpretations of user biometrics which motivates our work. But various issues like varied feature space, keystroke latency timings and human interpretability of the music complicate this mapping.

### 6.1.1 Key contributions of the section

Our key contributions are;

- ***Develop method to convert keystroke signature to musical signature:*** We present a method to map keystroke features to music notes which can be used as a musical signature. Using two modified functions to compute a note's duration and pitch, we are able to derive the musical equivalent of a keystroke signature. Our designed procedure to map keystroke features to musical signatures for a user is portable to other forms of behavioral biometrics, such as gait, swipes and wrist movements, with some modifications.
- ***Analyze inter-user and intra-user distances between music samples:*** We analyze the efficiency of music files for user verification using inter-user and intra-user distances between two key vectors; note-pitch and note-duration.

- ***Present human discretion results:*** Results from human subjects with little to no formal background in music, performing verification based on the music files of user are presented. Human classifiers were trained by listening to music files to verify users. User-wise accuracies, Type 1 errors (false rejects) and Type 2 errors (false accepts) are presented.

### **6.1.2 Related work**

The benefits of transforming information between the visual and aural senses has been studied in various contexts other than biometrics. Meijer [113] designed and evaluated a system that represented image information in form of sound. Inverse mapping (sound-to-image) mapping experiments showed convincing evidence for the preservation of visual information through the transformation. Kim [89] explored the other direction of information mapping by presenting techniques to represent sound in form of visual images. A few studies use such mapping techniques and extend existing systems to be more accessible to the visually impaired. Matta et al. [112] proposed a theoretical system that provided auditory image representations as an approximate substitute for vision, whereas Rigas and Memery [144] used both audio-visual stimuli to communicate information in browsing e-mail data. Both studies found auditory representation of data to be useful not only for the blind but also to maximize the volume of information communicated.

A host of other studies propose addition of audio cues to authentication systems. Saxena and Watt [152] explore various challenges and possible solutions for user authentication and device authentication in case of blind users. Goodrich *et al.* [68] investigate the use of

audio for human-assisted authentication of un-associated devices, their system, Loud-and-Clear, couples vocalization on one device and visualization on another for secure device pairing. Audio cues have also been used in systems to communicate a predetermined message to the visually impaired user. These audio cues can have various meanings, Wong *et al.* [185] use audio cues to signal the quality of the acquired image for face recognition when used by visually impaired users. Their study showed shorter acquisition times and improved rate of face detection for non-sighted users. Namin *et al.* [162] use a similar concept of audio cues to alert users about internet security threats with threat-sound pairs like; phishing-casting a fishing reel, malvertising-dropping a bomb, form filling-typing on keyboard.

The association of keystrokes to sound has been explored by few researchers, only in context of the acoustic emanations that occur while a user is typing. Zhuang *et al.* [197] show how keyboard acoustic emanations from 10 minute recordings can be used to attack and recover up to 96% of typed characters. Roth *et al.* [146] proposed keystroke sound as a modality of authentication of users in a continuous authentication scenario and discuss the shortcomings and possibility of better features in their work. In another attack focused work by Zhu *et al.* [196], off-the-shelf smartphones are used to record keystroke emanations. The authors use Time Difference of Arrival (TDoA) method and show that more than 72% of keystrokes can be recovered without any context-based information. A similar study by Liu *et al.* [103] performed better by recovering 94% of keystrokes with acoustic emanations and discrimination of mm-level position differences that help locate origin of keys on a keyboard. Another work by Roth *et al.* [148] investigated the discriminative power of these keystroke emanations, with an EER of 11% they conclude that there

is promising discriminative information in the keystroke sound to be further explored.

These works primarily focus on acquiring sound at the point of typing, which might not be audible or easily understandable to make meaningful interpretations by humans.

In a similar musical mapping work by Paul et al. [136] proposed a method to generate personalized music from DeoxyriboNucleic Acid (DNA) signatures of users. The number of Short Tandem Repeats (STRs) and the STR sequences were used as the units mapped to musical elements.

### **6.1.3 Details of the data collection**

The typing data was collected from 30 participants at our University after the IRB approval. The participants consisted of 13 females and 17 males, aged from 19 to 28. All participants were right-hand dominant and fluent in English. Twelve participants indicated that they were touch typists while the rest indicated to be visual typists. The participants performed the following activities on a desktop, with a standard QWERTY keyboard: multiple brief and interleaved sections of transcription, free-text typing, browsing and online shopping. This is a subset of SU-AIS BB-MAS dataset [83].

We focus on the transcription activities and only extract the data generated from the users while typing the phrase "this is a test" (hereinafter referred to as "test-phrase"). All users typed the test-phrase at many different points in their session, a minimum of 30 occurrences for each user were extracted from the data. We consider each occurrence of the test-phrase as one sample from the user.



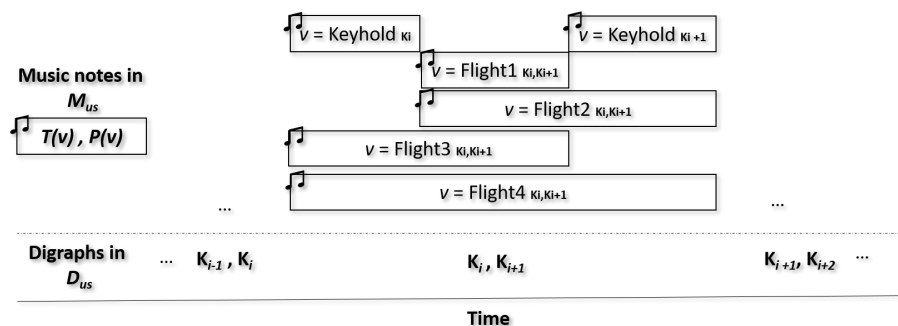


Fig. 6.1.: Music notes and their placement, generated from a digraph  $K_i, K_{i+1}$  using functions  $T(v)$  and  $P(v)$  for duration and pitch respectively.

#### 6.1.4 Music features

Keystroke data consists of keyhold times and inter-key (flight) latencies grouped by their keys of origin whereas music is generated from notes of different pitch and duration played in a certain pattern. Mapping data between these two very different modes of information is complicated as the magnitudes of the keystroke features, information of their different keys of origin, repeating key presses and many other such peculiarities cannot be expressed with simple equivalencies in the music domain.

Therefore, we shortlist the elements of music [187] that can be manipulated to create music from the keystroke features. We chose the following; *Melody*: the tune generated due to successive single notes affected by pitch and rhythm. *Harmony*: sound produced by two or more notes played simultaneously. *Rhythm*: combinations of sounds of varied length. *Pitch*: sound varied with the frequency of vibrations. *Tempo*: speed at which the music is played. By controlling the pitch and duration of musical notes, we can manipulate these five elements of music.

The MIDI protocol is a message-based communication between computer and equipment. The MIDI protocol was initially made to create polyphonic sound by using multiple musical devices once, linked with cable, in the music industry [61]. To create music that complies with MIDI standard we use *Ken Schutte MIDI Matlab Toolbox* [75]. The *matrix2midi* module takes a  $N * 6$  matrix and converts it to MIDI format which is then written to a MIDI file using the *writemidi* module.  $N$  rows of the matrix represent the notes (one for each note) and the 6 columns represent the *track number*, *channel number*, *note number* (midi encoding of pitch), *velocity* (volume), *start time* (seconds), *end time* (seconds) respectively. For simplicity, we set *track number* = 1, *channel number* = 1 (piano), and *volume* = 75 to be constants. By varying the pitch of a note and its duration we generate MIDI files  $M_{us}$  for each  $D_{us}$ .

### Mapping keystroke features to music

To generate  $M_{us}$  we compute their MIDI matrices of shape  $N * 6$ , we can denote  $M_{us}$  as:

$$M_{us} = [t\vec{n}_{us}, c\vec{n}_{us}, \vec{p}_{us}, \vec{v}_{us}, s\vec{t}_{us}, e\vec{t}_{us}] \quad (6.1)$$

Where  $t\vec{n}_{us}$  is track number,  $c\vec{n}_{us}$  is channel number,  $\vec{p}_{us}$  is pitch,  $\vec{v}_{us}$  is volume,  $s\vec{t}_{us}$  is start time, and  $e\vec{t}_{us}$  is end time and all the vectors are of the same length  $N$  (number of notes). As explained earlier,  $t\vec{n}_{us} = \vec{1}$ ,  $c\vec{n}_{us} = \vec{1}$  and  $\vec{v}_{us} = \vec{75}$ .  $\vec{p}_{us}$ ,  $s\vec{t}_{us}$  and  $e\vec{t}_{us}$  are mapped from  $D_{us}$  using our embedding technique to enhance the user-specific information held in  $D_{us}$ .

Each digraph  $\langle K_i, K_{i+1} \rangle$  in  $D_{us}$  (in alphabetic order) is mapped to music notes by converting its six associated keystroke values ( *Keyhold* of  $K_i$ ,  $K_{i+1}$  and their four *flight* values) with *duration function*  $T(v)$  and *pitch function*  $P(v)$ .  $T(v)$  is a simple scaling function, to scale the keystroke feature (in milliseconds) to practical music note duration (in seconds).  $P(v)$  is a modified form of MIDI Tuning Standard (MTS) which is specified in the MIDI protocol [17]. The two functions are shown below:

$$T(v) = v/100 \quad (6.2)$$

$$P(v) = 69 + 12 \log_2 \left( \frac{v}{440} \right) \quad (6.3)$$

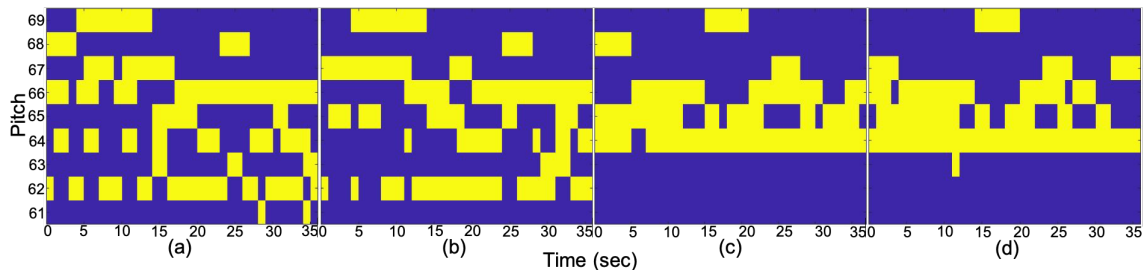


Fig. 6.2.: Examples of the piano roll plots that are obtained after mapping the keystroke features to the music features. We illustrate the piano roll plots of two test-phrase samples from two random users from our data-set, Figures 6.2(a) and 6.2(b) are from samples of user A and Figures 6.2(c) and 6.2(d) are from user B.

where  $v$  is the value from the  $X_f : v$  pairs in  $D_{us}$ . We substitute  $v$  in place of the frequency in the standard MTS equation. Since 440 Hz is a widely used standard *concert A* (musical note), equation (6.3) uses the term  $\log_2 (v/440)$  to compute the number of octaves above or below the *concert A*. This term is multiplied by 12 to compute the semi-

tones above the *concert A*. MIDI represents the *concert A* with integer 69 which is added for a MIDI compliant pitch number.

After computing duration and pitch of the notes, the notes are arranged similar to their occurrences over the duration of a digraph to obtain the vectors  $\vec{p}_{us}$ ,  $\vec{st}_{us}$  and  $\vec{et}_{us}$ . As *Flight4* translates to the longest duration, all other notes overlap with it at different points.  $\vec{et}_{us}$  is computed as  $\vec{st}_{us} + T(v)$ , notes corresponding to *Keyhold* $_{K_i}$ , *Flight3* $_{K_iK_{i+1}}$ , *Flight4* $_{K_iK_{i+1}}$ , have the same values in  $\vec{st}_{us}$  (same *start time*). *Flight1* $_{K_iK_{i+1}}$  and *Flight2* $_{K_iK_{i+1}}$  have the same values in  $\vec{st}_{us}$  equal to  $\vec{et}_{us}$  values of *Keyhold* $_{K_i}$ . *Keyhold* $_{K_{i+1}}$  has its  $\vec{st}_{us}$  value equal to the  $\vec{et}_{us}$  value from *Flight1* $_{K_iK_{i+1}}$ .

Figure 6.1 illustrates the mapping, *start time* and *end time* of music notes generated using digraphs from  $D_{us}$ . These notes (from all digraphs in alphabetical order) when played in a sequence produce a musical tune. Figure 6.2 shows a collection of piano roll plots which were generated for different samples of test-phrase using our procedure. The highlighted sections represent a played note. We can observe that plots 6.2a and 6.2b appear to be similar (similar sounding music) to each other. Both were generated from different samples by the same user. Figures 6.2c and 6.2d show the same for a different user. This example is representative of observations on our entire dataset.

### 6.1.5 Analysis on music from keystrokes

We performed inter-user and intra-user distance analysis and user-music verification using random forests, naive bayes and SVM. But as standard classifiers do not differentiate

between musical notes and other forms of data, we also perform verification experiments with three human-classifiers detailed below.

### 6.1.6 Inter-user and intra-user analysis

In each music file  $M_{us}$ , vectors  $\vec{tn}_{us} = \vec{1}$ ,  $\vec{cn}_{us} = \vec{1}$  and  $\vec{v}_{us} = \vec{75}$  are constant. Therefore we perform the inter-user and intra-user analysis using only the  $\vec{p}_{us}$ ,  $\vec{st}_{us}$  and  $\vec{et}_{us}$  vectors. As order of the notes in all music files are same, vectors  $\vec{st}_{us}$  and  $\vec{et}_{us}$  can be simplified to a single vector  $\vec{d}_{us} = \vec{et}_{us} - \vec{st}_{us}$ .  $\vec{p}_{us}$  denotes the pitch of the notes and  $\vec{d}_{us}$  denotes their duration. We chose Canberra distance as it is most suitable when dealing with vectors.

The Canberra distance between two vectors  $\vec{a}$  and  $\vec{b}$  is given by:

$$c(\vec{a}, \vec{b}) = \sum_{i=1}^n \frac{|a_i - b_i|}{a_i + b_i} \quad (6.4)$$

Figure 6.3 shows the density functions of the inter-user and intra-user distances from all music files. Figure 6.3a is plotted with distances using  $\vec{p}_{us}$  while figure 6.3b is using  $\vec{d}_{us}$  from all the music files respectively. We observe that the density curves for intra-user distance falls majorly towards the left, implying lesser intra-user differences in music, for both cases. In contrast, the density curves for inter-user distances are fall towards the right with higher distance values. We can also observe the overlapping regions between the intra-user and inter-user density curves is small. Small intra-user distances, large inter-user distances and small overlap among these curves are all desirable qualities for user

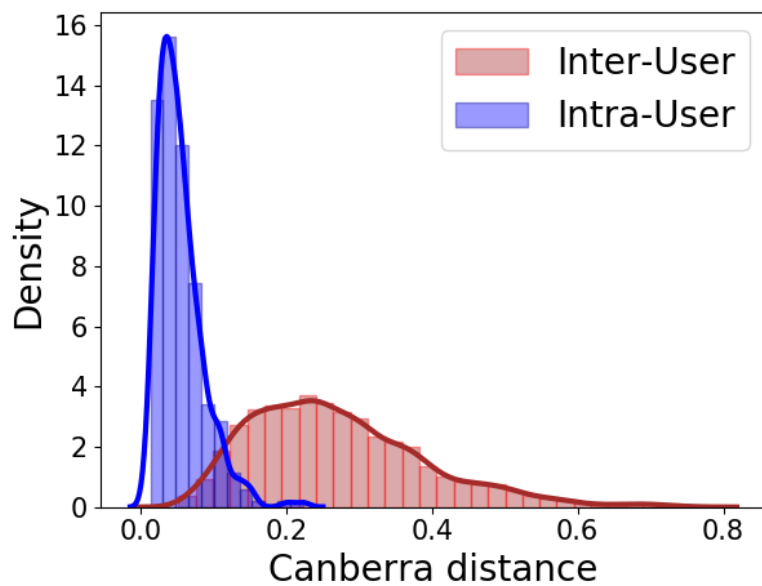
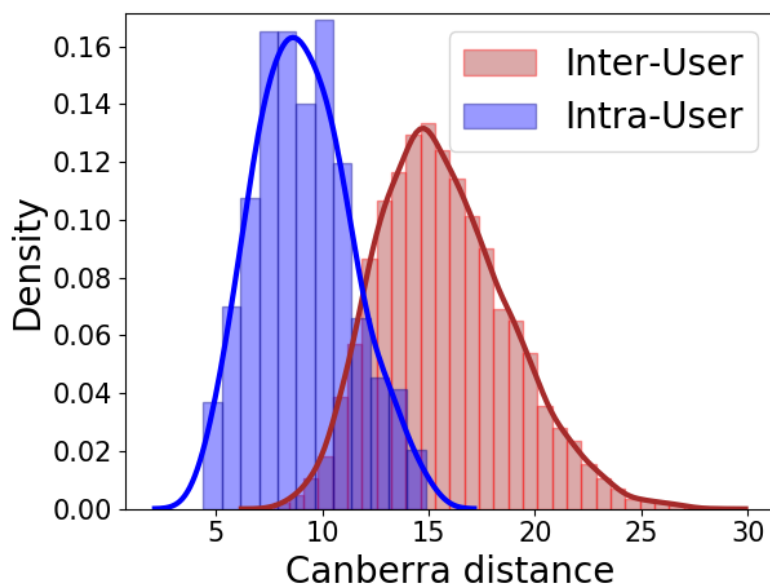
(a) Distances using note-pitch vectors  $\vec{p}_{us}$ .(b) Distances using note-duration vectors  $\vec{d}_{us}$ .

Fig. 6.3.: Plot of density functions for inter-user and intra-user Canberra distances of the note-pitch vectors (6.3a) and note-duration vectors (6.3b) between all music files.

verification. These qualities imply that music files of a user are fairly separable from other users.

### Verification experiments with standard classifiers

Even though We use the note-pitch vector ( $p_{us}^{\rightarrow}$ ) and the note-duration vector ( $d_{us}^{\rightarrow}$ ) as feature vectors for the music files. For the verification experiments, we use three different classifiers; Random Forests, Naive Bayes and SVM. Due to our dataset consisting of 30 music files for each user, we run the experiments with two different configurations; two-fold and three-fold cross validation. For each session 30 imposter samples are sampled randomly from users other than the genuine user.

Random forest classifier with five trees, maximum depth was restricted to five and number of child nodes was restricted to two. GINI impurity was used for the split criterion.

In SVM classifier we use a RBF kernel, penalty parameter = 1 and gamma = 0.01. The Gaussian Naive Bayes (GNB) classifier implements the Gaussian Naive Bayes algorithm as shown by the following equations:

$$\hat{y} = \underset{y}{\operatorname{argmax}} P(y) \prod_{i=1}^n P(x_i|y) \quad (6.5)$$

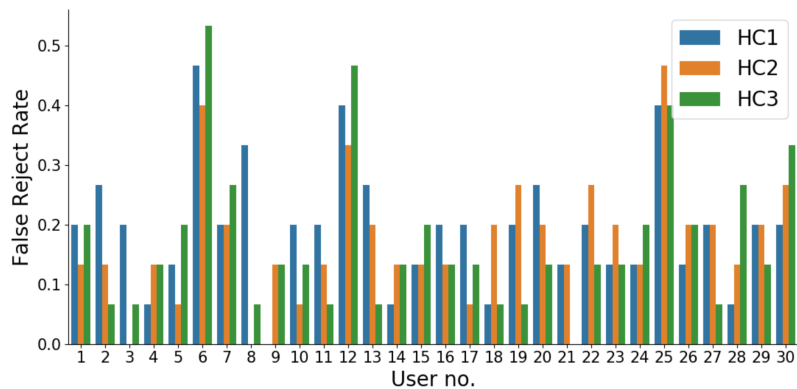
$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right) \quad (6.6)$$

where,  $y$  is the class,  $\hat{y}$  is the predicted class,  $x_1, x_2, \dots, x_n$  are the features,  $\sigma_y$  and  $\mu_y$  are the standard deviation and mean estimated using maximum likelihood. Table 6.1 presents the results of our verification experiments. We observe that the Naive Bayes and SVM classifiers perform slightly better with more number of training instances. However, all three classifiers performed similarly with high accuracy between 89% to 96%. The False

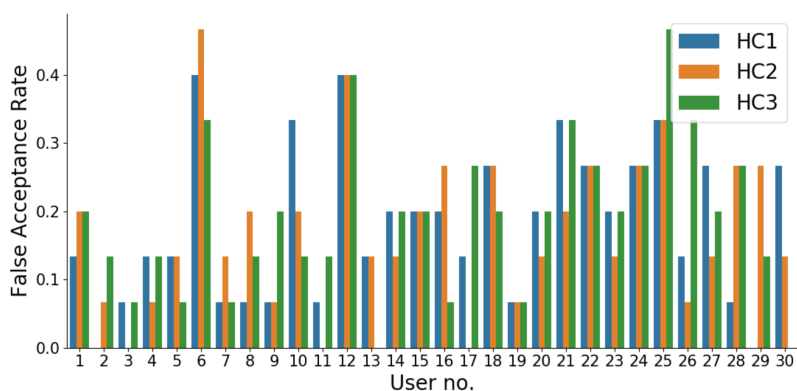
Table 6.1: The average FAR, FRR and Accuracy; for the three standard classifiers with two-fold and three-fold cross validation experiments (on the left) and for human classifiers (on the right) on user verification.

Metrics Average	Standard Classifiers			Human Classifiers		
	<i>Random Forest</i>	<i>Naive Bayes</i>	<i>SVM</i>	<i>HC1</i>	<i>HC2</i>	<i>HC3</i>
FAR	0.06 ± 0.05	0.12 ± 0.07	0.14 ± 0.08	0.18 ± 0.11	0.20 ± 0.10	0.17 ± 0.11
FRR	0.09 ± 0.07	0.10 ± 0.09	0.14 ± 0.10	0.18 ± 0.10	0.19 ± 0.11	0.17 ± 0.12
Accuracy	0.92 ± 0.0	0.89 ± 0.06	0.89 ± 0.07	0.81 ± 0.09	0.83 ± 0.10	0.81 ± 0.12
FAR	0.06 ± 0.07	0.05 ± 0.06	0.04 ± 0.07	Overall		
FRR	0.03 ± 0.05	0.03 ± 0.05	0.04 ± 0.08	0.18 ± 0.11		
Accuracy	0.96 ± 0.04	0.94 ± 0.05	0.96 ± 0.05	0.81 ± 0.10		

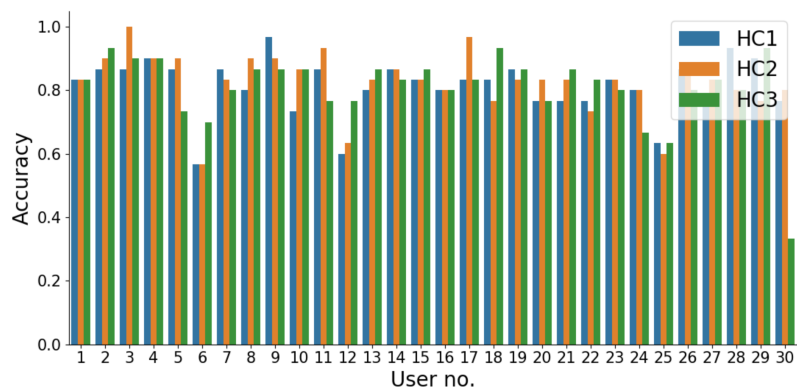




(a) False rejection rates or Type-1 error.



(b) False acceptance rates or Type-2 error.



(c) User-wise accuracies for verification.

Fig. 6.4.: Results from the Human-Classifier (HC) based verification experiments.

Rejection Rates (FRR) and False Acceptance Rates (FAR) were low ( $\leq 10\%$ ) in all cases except for two-fold experiments with Naive Bayes and SVM.

As these classifiers do not differentiate between music and other numerical data, these results do not highlight the merits of converting a user's keystroke data into music. Therefore, a human classifier based experiments were carried out, detailed below.

### **Verification experiments with human classifiers**

The true test and application of our work is to see if an average human can differentiate between the music generated from one user's typing sample to another. We recruited three volunteers (hereinafter referred to as human-classifier or HC) with little to no formal education in music. Each experiment session had a training phase and a testing phase. A user number was selected for each session and all the samples from that user were labeled "genuine" for the session. In the training phase the HC was made to listen to 15 music files from the genuine user. In the following testing phase a set of 30 music files, consisting of 15 genuine (not used in training) and 15 imposter (randomly selected from other users) files were played one after the other. At the end of each file HC classified it as either genuine user or as an imposter. The classification decisions were recorded and analyzed.

Figure 6.4 and Table 6.1 summarize the results of our verification experiments carried out with three HCs. Figures 6.4a and 6.4b show the Type-1 and Type-2 errors committed by the HCs respectively. Type-1 error is the case where a HC falsely rejected a genuine music sample (rejection of a true null hypothesis). Whereas, Type-2 error is the case where a HC falsely accepted an imposter's music sample (failure to reject a false null hypothesis). Figure 6.4c shows the user-wise accuracy of the HCs in the verification task. We observe

that verifying a few users was challenging for all three HCs, especially users 6,12 and 25, reflected in the high FAR and FRR for these users. For all other users, the FAR and FRR values are low, within range of 10% to 15% in most cases. In a few cases the FAR and FRR values are 0, indicating a perfect classification by the HC. The accuracies shown in the figure 6.4c reflect similarly with most being in the range of 75% to 85% while being low on users 6, 12 and 25. Overall, all three HCs could easily verify the music files of users with high accuracies for most users.

### **6.1.7 Conclusion and future work**

Our work shows that information from keystroke dynamics can be translated to other representations that maintain or enhance the human interpretability of the data. A theoretical system to convert keystroke features to the aural sense has been proposed. Information on KD is conveyed through auditory music files. We show that user-specific music files exhibit high inter-user distance and low intra-user distance which is a desired quality in a feature vector to be used for authentication. Verification experiments with standard classifiers (Random Forests, Naive Bayes and SVM) show that these music files can be verified with high accuracies despite treating music as any other numerical data. Experiments with different devices, types of text and music fluency of the human classifiers are part of our future research direction.

Results from human-classifiers reveal that a user's keystroke behavior can be converted to music which is humanly verifiable. The musical melody created from each user's keystroke data using our approach has a clearly distinguishable tune unique to a user. The approach

and findings of this work can be used in a variety of ways such as; a new mode of second-factor user authentication, a complementary form of data presentation for audible user specific keystroke signatures. The concept of mapping biometrics features to music can be reused with some modifications to suit other forms of biometrics such as gait, touch, swipe and fingerprints to name a few.

## **6.2 DoubleType: Authentication Using Relationship Between Typing Behavior on Multiple Devices**

Authentication using Keystroke Dynamics (KD), has customarily focused on one device at a time, either desktop or phone. It is imperative that authentication systems adapt to an environment where the users consistently switch between multiple devices. We use the typing behavior of users on different devices, extract the relationship between them and show that these relationships can be used to authenticate users in a multi-device environment when a user switches between devices. We design an authentication system for three scenarios, using the relationship between typing behaviors on a) desktop and phone, b) desktop and tablet, and c) tablet and phone. We find that these are highly separable for individuals with data from 70 users. With Gaussian Naive Bayes (GNB) and Random Forests (RF) classifiers, we found the accuracies for verification to be very high. Using GNB we achieved mean accuracies of 99.15%, 99.23% and 98.72% for relationships between desktop-phone, desktop-tablet and tablet-phone respectively. RF classifiers performed similarly with mean accuracies of 99.31%, 99.33% and 99.12% for relationships between desktop-phone, desktop-tablet and tablet-phone respectively.

User authentication has always been a challenging domain. Authentication mechanisms have evolved gradually with introduction and popularity of new devices into the consumer market. While, Personal Identification Number (PIN) and passwords are some of the classical authentication methods, behavioral biometrics brings a variety of authentication mechanisms to the table [10, 34, 134]. Keystroke Dynamics (KD) is a form of behavioral biometrics which uses a user's typing behavior to perform identification or verification. KD has shown promising results as a security measure throughout its research [54, 120, 179]. KD based authentication has been extensively studied in two main categories of devices, devices with physical keyboards like desktops or laptops [29, 33, 70, 81], and touchscreen devices like smart-phones [62, 84, 108].

As more and more people are getting accustomed to multi-device environments, it would be a step in the right direction to adapt the field of KD to this inevitable future. It is not a rare occurrence for a person of this generation, to work or use multiple devices in an interleaving fashion.

In this work we explore how the typing behavior of a user on one device can be related to the typing behavior of the same user on another device and if it is unique enough from different users using those devices. We also describe an approach to use this relationship as means of verification. As multi-device multi-user environments become the norm, it is critical for future authenticating systems to have provisions to check such multi-device behavior relationships. Verification using relationships involving multiple devices also adds a protective layer if one of those devices are compromised.

### 6.2.1 Key contributions of the section

Our key contributions are;

- ***Develop and extract multi-device typing behavior relationship:*** We show that the *relationship* between a user's typing behavior on two devices is fairly unique and can be used for authentication in multi-device environments. We posit that, because typing samples from two different devices are used in our authentication system, this system would be harder to spoof in comparison to systems that use typing samples from a single device. In this section cover all three scenarios of user's typing behavior relationships, a) desktop-phone; b) desktop-tablet; and c) tablet-phone.
- ***Present results with scaling number of users:*** We show that the proposed system is accurate and scalable. We vary the number of users 10 users and increment up to 70 users and all vital metrics such as, Precision, Accuracy, Specificity and Sensitivity stay consistent in the range of 95% to 99.5%, implying highly scalable approach.

### 6.2.2 Related work

Authentication using KD has been studied from various perspectives for quite sometime now. From early works of Gaines [66], the primary focus of KD has been on the way an individual uses a keyboard interface. A majority of research in authentication using KD is based around a typical desktop setup, where a user is using a standard keyboard, pressing and releasing a sequence of keys. Some of the early works [33, 81, 129, 131] provided insights of how keystroke dynamics could be used in an authentication system. Though the

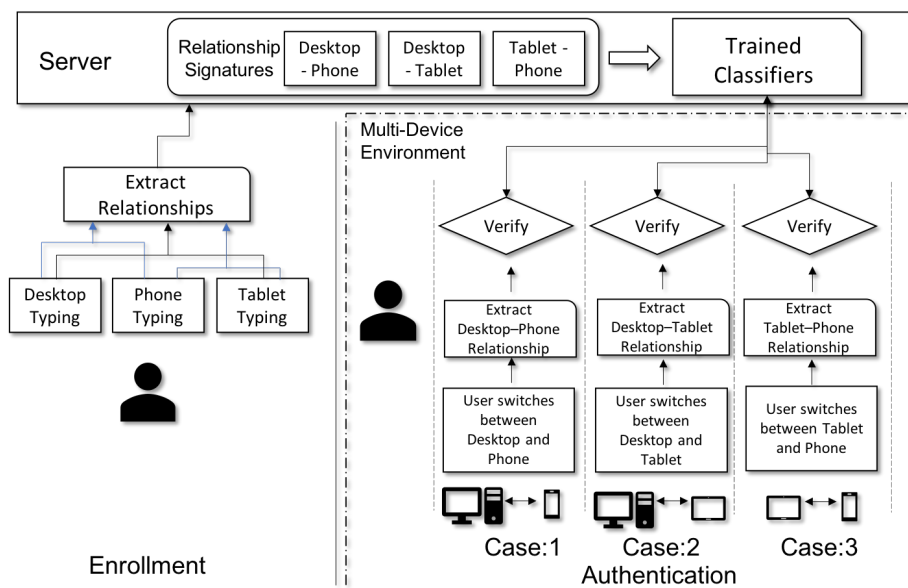


Fig. 6.5.: An overview of the authentication system.

number of participants in early studies were very small, often between ten and twenty, and the type of text was usually fixed text of small lengths, their findings had a huge impact nonetheless. Various features like key-hold durations, inter-key latencies [81] between digraphs [95], trigrams, and their effects on classification systems were studied. In Obaidat and Sadoun's [129] work it was shown that a combination of key-hold time and flight time features resulted in better classification of users.

A detailed study, with a considerable user population, on free text in authentication by KD was carried out by Gunetti and Pacardi[70]. In their earlier work by Bergandano [29], the authors used the same sampling text for all individuals and allowed typing errors. However, in a study by them later [70], use of free text in KD research assured higher usability as compared to using only fixed text.

As mobile phones and smart-phones became more popular, authentication systems have also been proposed using the KD on these devices. Clarke [46] in their feasibility study

of KD on mobile handsets, found KD to be promising as part of a larger hybrid authentication algorithm. Rodrigues [145] proposed a KD based access control for devices with numerical keyboards, which was the case of early mobile phones. Clarke and Furnell, in their successive works [44, 48], described detailed frameworks and presented analysis on using KD for authenticating mobile phone users. In their works, the authors focused on three scenarios; entry of 11-digit phone numbers, entry of 4-digit PINs and entry of text messages.

### **How our work is different?**

Most KD based authentication systems limit their focus to a particular device, either a desktop or a phone. Researchers use the typing behavior of a user on either one of these devices to authenticate the user on the same device. We look at a multi-device environment and focus on the typing behavior of users on multiple devices to design out authentication system. We find that the relationship of typing behavior of a person on two devices is quite unique and can be used for authentication. We chose the three most popular devices; desktop, phone and tablet. Our dataset consists of a considerable number of users, performing tasks on all of these devices. The richness of this collected dataset itself enables us to pursue different paths of analysis which sets our work apart from the rest. Efforts are underway to make our dataset available publicly to benefit all researchers.

We relate the typing behavior of same user on different devices and test the separability of this relationship from different users typing on these devices. As we have seen researchers combine different modalities in our literature search, our work can be viewed as an at-



tempt to combine the same modality (typing behavior or KD) from different devices for authentication, which has not been explored in KD base authentication.

### 6.2.3 Overview of the authentication system

Figure 6.5 presents an overview of our method. It is applicable in environments where a user switches between two or more devices while going about their tasks. Our authentication system can be deployed either separately or as a second layer, in conjunction with other authentication systems on the individual devices. During the enrollment phase, multiple typing samples from the users typing activity on each pair of device (Desktop-Phone, Desktop-Tablet, Tablet-Phone) is used to extract the *relationship – features* (detailed in section IV). These relationships act as signatures for the typing behavior of a user on these pairs of devices. Only the relationship between the samples need to be stored on the server. When a user switches to another device, verification samples from both the devices are used to build a relationship (see Section IV) which is then input as a feature vector to a classifier to verify the identity on the switched device.

Our conjecture is that even if any device is compromised or both devices simultaneously are compromised, it will be difficult to compromise the relationship between the devices for the specific individual because the relationship-signatures build on dictionary values containing statistics of unigraphs and digraphs, whereas the individual verifiers on devices will use standard set of features, which will require reverse engineering raw values from features. Thus this method provides an important security layer.

Table 6.2: Summary of the data collection.

<b>Device</b>	<b>Desktop</b>	<b>Tablet</b>	<b>Phone</b>
<b>Tasks</b>	Transcription (Fixed Text) Browsing (Free Text) Q & A (Free Text)	Transcription (Fixed Text) Q & A (Free Text)	Transcription (Fixed Text) Q & A (Free Text)
<b>Approx. Duration</b>	55 mins.	30 mins.	30 mins.
<b>Approx. Keystrokes per participant</b>	12500	9000	10000
<b>Keyboard Interface</b>	Standard QWERTY Dell Kb212-b	Touchscreen QWERTY HTC Nexus 9	Touchscreen QWERTY Samsung S6 / HTC One
<b>Gender</b>	Male : 38      Female : 32		
<b>Time Spread</b>	Approx. 2 months		
<b>Age group</b>	19 years to 35 years		

#### 6.2.4 Details of the data collection

As, at the time of this study, keystroke data of the same person typing on different devices was not publicly available, we had to make our own data collection effort. Data collection was carried out after the IRB approval from our university. The data from 70 users is used in this study. The population consisted of 38 male and 32 female participants. They were aged between 19 and 35 years. The data collection was designed to emulate common activities on three devices; a desktop, a tablet and a phone. The interfaces for keystroke input on the devices were: a standard QWERTY keyboard for the desktop, the touchscreen QWERTY keypad of HTC-Nexus-9 for tablet, the touchscreen QWERTY keypad of Samsung-S6 or HTC-One for phone. The typing activities required to user to type a mix of both, free text and fixed text, which closely mimics the text typed in a real-life scenario.

The participants completed activities in the following order: a) desktop, b) tablet and c) phone. All keystroke events and their timestamps were logged during these activities. The users took about 55 minutes to complete the tasks on desktop and 30 minutes each on tablet and phone. Each participant made approximately 12,500 keystrokes on the desktop, 9,000 keystrokes on the tablet and 10,000 keystrokes on the phone. The data collection took around 2 months to complete. The data is a subset of SU-AIS BB-MAS dataset [83] that is shared publicly for the benefit of the research community. A summary of our data collection efforts are presented in table 6.2.

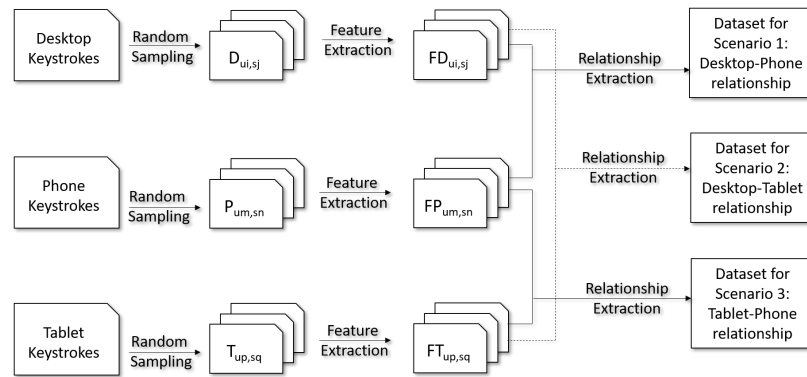


Fig. 6.6.: Data preprocessing and formation of datasets from the relationship between typing behavior on two devices.

## 6.2.5 Methodology and experiments

**Outlier Removal:** For the detection and removal of outliers, we use a simple filter to remove any instances of features which have a value of two seconds or more. We assume that these were caused by pauses, where the user was either thinking or receiving instructions during the data collection.

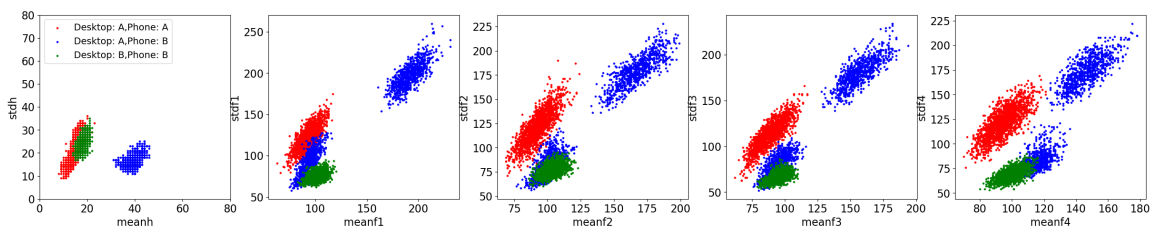


Fig. 6.7.: Illustration of the feature values from two random users selected from our dataset for scenario 1: *relationship – features* for Desktop and Phone.

### Data preprocessing and relationship extraction

The data preprocessing procedure is outlined in Figure 6.6. In the first step of our data preprocessing, we extract 40 random samples from each user’s data from each device. Each sample is made from the keystroke events of approximately 300 characters (we

chose samples with 300 characters as it is approximately equal to the character limit of a tweet: 280 characters (Twitter is a popular social platform where people express their thoughts within the said character limit). The extracted groups of samples can be denoted as  $D_{u_i, s_j}$ ,  $P_{u_m, s_n}$  and  $T_{u_p, s_q}$ , where  $D$ ,  $P$ ,  $T$  are samples from desktop, phone and tablet respectively.  $u_i$ ,  $u_m$  and  $u_p$  are the user, i, m and p respectively, which range from 1 through 70.  $s_j$ ,  $s_n$  and  $s_q$  are the sample number, where j, n and q ranges from 1 through 40. For example,  $D_{u_5, s_{10}}$  represents the keystroke sample from desktop for user number: 5 and sample number: 10. We then extract the keystroke features that were discussed above, to form the feature template for each sample, which can be denoted as  $FD_{u_i, s_j}$ ,  $FP_{u_m, s_n}$  and  $FT_{u_p, s_q}$ , where  $FD$ ,  $FP$ ,  $FT$  are feature templates from desktop, phone and tablet samples respectively. All other aspects of the denotations are same as explained above. Each feature template is comprised of five dictionaries, one for each feature. These dictionaries can be denoted by  $D_{KH}$ ,  $D_{F1}$ ,  $D_{F2}$ ,  $D_{F3}$  and  $D_{F4}$  for the features keyhold, flight1, flight2, flight3 and flight4 respectively. Each dictionary is made of multiple *key : value* pairs. The *keys* are Uni-graphs, as in the case of  $D_{KH}$ , or Di-graphs, as in the case of  $D_{F1}$ ,  $D_{F2}$ ,  $D_{F3}$  and  $D_{F4}$ . The *value* is average of the feature for the *key* occurring within the corresponding keystroke sample. Therefore, continuing our previous example, the feature template of the keystroke sample  $D_{u_5, s_{10}}$ , would be  $FD_{u_5, s_{10}} = [D_{KH}, D_{F1}, D_{F2}, D_{F3}, D_{F4}]$

### Relationship extraction

To use relationship of a user's typing behavior on different devices as an authentication criteria, for two feature templates say  $FX$  and  $FY$ , from two different devices (such that  $FX$  and  $FY$  are both not from same group  $FD$ ,  $FP$  or  $FT$ ), we extract the following *relationship – features* from them:

- a.  $Avg.KH, Std.KH$  : The average and standard deviation, of absolute difference between corresponding *key : value* pairs in  $D_{KH}$ .
- b.  $Avg.F1, Std.F1$  : The average and standard deviation, of absolute differences between corresponding *key : value* pairs in  $D_{F1}$ .
- c.  $Avg.F2, Std.F2$  : The average and standard deviation, of absolute differences between corresponding *key : value* pairs in  $D_{F2}$ .
- d.  $Avg.F3, Std.F3$  : The average and standard deviation, of absolute differences between corresponding *key : value* pairs in  $D_{F3}$ .
- e.  $Avg.F4, Std.F4$  : The average and standard deviation, of absolute differences between corresponding *key : value* pairs in  $D_{F4}$ .

Our final dataset consists of the following 10 feature columns:  $Avg.KH, Std.KH, Avg.F1, Std.F1, Avg.F2, Std.F2, Avg.F3, Std.F3, Avg.F4$  and  $Std.F4$ . Apart from these feature columns, the columns  $UserIDs$ , who typed the text samples are used to dynamically create a class label column.

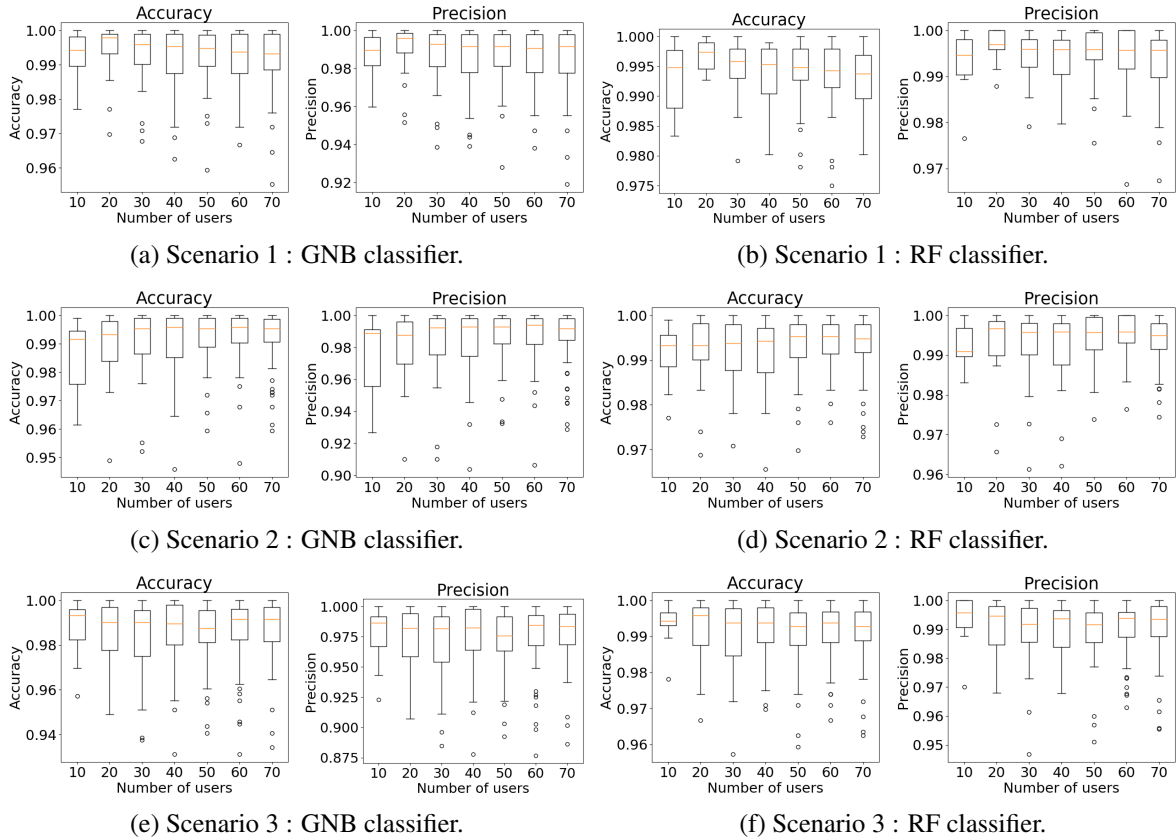


Fig. 6.8.: Performance of the two classifiers for all three scenarios, Desktop-Phone (6.8a and 6.8b); Desktop-Tablet (6.8c and 6.8d); and Tablet-Phone (6.8e and 6.8f) relationship.

- Scenario 1; **Desktop-Phone** relationship: We extract the *relationship – features* between  $FD_{ui,sj}$  and  $FP_{un,sm}$  where  $i$  and  $n$  range from 1 to 70 (user number) and  $j$  and  $m$  range from 1 to 40 (sample number).
- Scenario 2; **Desktop-Tablet** relationship: Similar to scenario 1, but, with  $FD_{ui,sj}$  and  $FT_{up,sq}$ .
- Scenario 3; **Tablet-Phone** relationship: Similar to scenario 1, but, with  $FT_{up,sq}$  and  $FP_{un,sm}$ .

Figure 6.7, illustrates the *relationship – features* extracted between two random users from our dataset with the help of scatter plots for scenario 1. In this illustration we can

see different clusters of feature values. The different clusters signify if the same user was using both the devices in the given scenario, or if different users were using the devices.

## Experiments and classifiers

We perform verification experiments for the three scenarios using their corresponding datasets. We assign a class label to each instance in the dataset depending on the user number being verified. If the genuine user for a verification round is user  $G$  then, only the instances that have both  $user_1$  and  $user_2$  as  $G$  are assigned the class label of "1" (genuine); the rest are assigned "0". This implies that the relationships coming from the same user  $G$ , typing on both devices is considered a genuine instance for that round. The fields  $user_1$  and  $user_2$  are dropped and only the *relationship – features* are used to train and test the classifiers. We use two classifiers to perform verification from the scikit-learn library using python for programming. The first classifier, is a Gaussian Naive Bayes (GNB) classifier and Random Forest (RF) classifier. We use 10 trees in the RF with the Gini impurity for the split criterion. We also limit the maximum depth of the trees to be five. For both the classifiers we balance the genuine and impostor classes to have approximately equal instances and use a 70% of the data for training and the remaining 30% of the data for testing. To analyze the scalability of the approach, for all three scenarios, we start with the data from only 10 users, gradually incrementing the size in 10s until we finally include the data from all 70 users. The results of our experiments are discussed in the following section.



### 6.2.6 Results and discussion

We use Accuracy and Precision as the key metrics to measure the performance of both the classifiers. The mean performance values for verification in Scenario 1: Desktop-Phone relationship, are presented in Figures 6.8a and 6.8b. Both the classifiers perform consistently well. Even as the number of users in the system is increased gradually from 10 through 70, we observe that all the mean values for the metrics remain stable. Accuracies on both the classifiers are very high with above 99% accuracy throughout. Precision of the RF classifier is better than the precision of the GNB classifier within a range of 1%, but are still satisfactorily high on both classifiers nonetheless. The plots reaffirm our observation that all the performance metrics stay stable even with the increase in the number of users for both the GNB classifier (6.8a) and the RF classifier (6.8b). Figures 6.8c and 6.8d summarizes the results verification in Scenario 2: Desktop-Tablet relationship, both the classifiers perform similar to Scenario 1. With high Accuracy and Precision rates for all sizes of the user population that were considered. Even at the maximum user population of 70, we see that the average accuracy and precision values are 99.23% and 98.64% for the GNB classifier and 99.33% and 99.38% for the RF classifier respectively. The summarized results for Scenario 3: Tablet-Phone relationship, is shown in Figures 6.8e and 6.8f, similar to the previous two scenarios discussed, the values for Accuracy and Precision are very high for all sizes of user population in our experiments. For the final user population of 70, both the classifiers perform satisfactorily. We also observe that in case of the GNB classifier, the average accuracy and precision values are marginally smaller than the values for the RF classifier within a range of 1.0%.

### 6.2.7 Conclusion and future work

We conclude that the relationship between the typing behavior of users on different devices is fairly unique to an individual. We find from our study on 70 users and 3 devices (desktop, phone and tablet), that the relationship between the typing behaviors perform outstandingly for verification. Previous research in KD based authentication has focused exclusively on single device environments. Our results show that, KD based authentication from relationships between the typing behavior of users on multiple devices can be considered in a multi-devices environment. Some of the applications of this work could be authentication of users in online courses or exams and employee authentication in an office environment, or as a second layer of authentication.

We speculate that a user's typing behavior on individual devices may be easier to mimic and breach when compared to breaching the relationship of the typing behavior on two devices. This approach can be likened to a two-factor authentication approach, where the user has to type on two different devices and the relationship between these typing samples is tested.

This study leads to many other intriguing questions, such as: are there other relationships between the typing behaviors on different devices which may improve the results of authentication? What other activities on multiple devices, other than typing, can be used to form a multi-device authentication system? Can text samples with smaller character limit yield similar results? These are areas we plan to explore in future research.

## 7. SUMMARY

We share and provide the details of our large behavioral biometrics dataset for typing, gait and swiping activities of the same user on desktop, tablet and phone. The availability of the data on different devices for the same person makes our dataset unique; and with data from 117 participants, also one of the largest. With this dataset researchers can try to explore questions that were not possible with previously available datasets such as; *”Does the typing of an individual on desktop reveal their typing on a tablet or phone? and vice versa”* ; *”Can a person’s demographics like age, height, etc., be predicted from the data of typing, gait or swiping activity on any of the devices?”*; to name a few.

Our experiments show that, in the case of keystrokes, gait and swipes using desktop, tablet and phone, it would be wrong to assume an underlying normal distribution. Low values of  $p$  from our non-parametric normality tests across activities and devices show that researchers in behavioral biometrics must not assume the data to be from a Gaussian distribution to get better and more accurate insights. However, upstairs and downstairs activity data, showing higher percentages of samples where an underlying normal distribution cannot be discarded is intriguing and further research is needed to establish why this occurs. Knowing that the data does not follow normal distribution leaves the discussion incomplete, which can only be completed by learning alternate ways to handle a non-normal dataset. Our results question the common assumption that the data in behavioral biometrics follows a normal distribution. We have discussed the implications and

alternate approaches for such a scenario. We hope that, insights from our work help future researchers to make the right choices in terms of data models, transformations and classifiers to achieve better results and make correct interpretations.

The proposed word-specific features perform much better at user identification on all devices. Conventional features, especially KeyHold does not provide user separation to a desired level. We considered the subset of proposed features that offered higher discriminability, like WordHold, AvgFlight1, AvgFlight2, AvgFlight3, AvgFlight4, evaluated them with classifiers and drew comparisons with conventional features (Section 3.9).

These classifiers show competitive accuracies on all devices. Mathematical insights for this improvement in performance are drawn (Section 3.10.1). We also note that these features in general perform much better on hand-held devices. We speculate that user's style of holding devices and patterns such as, short bursts of typing followed by pauses between words might be some of the reasons (Section 3.10.2). Analysis of the word-based impact factors reveal that four or five character words, words with about 50% vowels, and those that are ranked higher on the frequency lists might give better results for the extraction and use of the proposed features (Section 3.10.3) for user identification.

We raised an intriguing question: "*Can the typing behavior of a user reveal if the typing activity is malicious or benign?*". We conclude that the typing behavior of a user can reveal if the typing activity being done is benign or malicious. Although, the keystroke features that have been popularly used for user identification or verification are not suitable for this task. We proposed a different set of features using which the origin of a text sample (whether malicious or benign) could be determined with high levels of accuracies. We observe that behavior of keystroke timings and frequencies of certain keys like *Space*,

*Enter* and *Punctuation* keys can be used to reveal the nature of typing activity. Using our proposed features we could achieve accuracies as high as 97% and Type 1 and Type 2 error rates of less than 3%. We show that keystroke analysis can be used to determine the nature of typing activity, thereby assessing the threat levels of a system. However, we understand that keystroke analysis would have to be used in conjunction with other technologies to obtain a more robust and secure system.

## APPENDICES

## **A. ADDITIONAL DETAILS OF DATA COLLECTION**

### **A.1 Cognitive Loads[35]**

<b>Task</b>	<b>Level</b>	<b>Required activity</b>
Remember	1	Retrieve knowledge from long-term memory to explain
Understand	2	Explain, summarize or interpret
Apply	3	Apply, execute or implement
Analyze	4	Organize or break material into constituent parts
Evaluate	5	Critique or make judgments based on criteria
Create	6	Generate, plan or put elements together

### **A.2 Examples of Free text questions on desktop**

- List some of the things that you like about Syracuse University.
- Which internet browser do you typically use (e.g, Google Chrome, Internet Explorer, Mozilla Firefox, etc.)?
- What improvements would you like to see in that browser?
- If you were to draw a picture of Syracuse University, what objects would you include in it?
- What is your favorite vacation spot? Why do you like to visit there?
- Give step-by-step driving directions to your favorite restaurant in the Syracuse Area, starting from your dorm room/ home.
- Discuss step-by-step instructions for making your favorite type of sandwich. Write them so that the person who has never done this before can follow your instructions.

### **A.3 Examples of Free text questions on tablet**

- What is your ideal job after graduation? Why?
- Why did you decide to attend Syracuse University?
- Re-read Question #2 (from the Multiple Choice Questions section) and the responses. Which response do you feel is least applicable to you and why?
- Review Question #6 (from the Multiple-Choice Questions section) and the answer that you chose. Why did you select your answer?
- If Question #6 (from the Multiple-Choice Questions section) was changed to read "If some mangoes are golden in color and no golden-colored things are cheap", which answer would be correct and why?

#### A.4 Examples of Free text Questions on phone

- Of the courses you've taken in college, which was your favorite and why?
- Think about a class that you did not enjoy. What improvements would you like to see to make the course better?
- Re-read Question #2 (from the Multiple Choice Questions section) and the responses. Which response do you feel is least applicable to you and why?
- Do you intend to pursue an advanced degree (e.g., Master's or Ph.D. )? Why or why not?
- Review Question #7 (from the Multiple-Choice Questions section) and the answer that you chose. Why was the rule you found/why did you select your answer?

#### A.5 Transcription Sentences

- "this is a test to see if the words that i type are unique to me. there are two sentences in this data sample."<sup>1</sup>
- "second session will have different set of lines. carefully selected not to overlap with the first collection phase."<sup>1</sup>

---

<sup>1</sup>The transcription sentences were selected based on two criteria: (1) inclusion of many frequently used words in the Oxford English Corpus, and (2) encouraging typing activity on both hands (on both sides on the keyboard). Transcription sentences were typed in lower case.



## LIST OF REFERENCES

## LIST OF REFERENCES

- [1] Abdella Zidan Amhemad. Effect of non normality on statistical control charts. In *2010 International Conference on Networking and Information Technology*, pages 512–515, June 2010. doi: 10.1109/ICNIT.2010.5508459.
- [2] S. Adewale. *A STATISTICAL ANALYSIS TO DETERMINE THE DISTRIBUTION AND PATTERN FOR AN INSURANCE HEALTH CLAIMS*. PhD thesis, 06 2017.
- [3] R. Adinehnia, N. I. Udzir, L. S. Affendey, I. Ishak, and Z. M. Hanapi. Effective mining on large databases for intrusion detection. In *2014 International Symposium on Biometrics and Security Technologies (ISBAST)*, pages 204–207, Aug 2014. doi: 10.1109/ISBAST.2014.7013122.
- [4] A. A. Ahmed and I. Traore. Biometric recognition based on free-text keystroke dynamics. 2014. *IEEE Transactions on Cybernetics*, VOL. 44, NO. 4.
- [5] A. A. E. Ahmed and I. Traore. Anomaly intrusion detection based on biometrics. In *Proceedings from the Sixth Annual IEEE SMC Information Assurance Workshop*, pages 452–453, June 2005. doi: 10.1109/IAW.2005.1495997.
- [6] K. Ali, A. X. Liu, W. Wang, and M. Shahzad. Keystroke recognition using wifi signals. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, MobiCom '15*, pages 90–102, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3619-2. doi: 10.1145/2789168.2790109. URL <http://doi.acm.org/10.1145/2789168.2790109>.
- [7] M. L. Ali, J. V. Monaco, C. C. Tappert, and M. Qiu. Keystroke biometric systems for user authentication. *Journal of Signal Processing Systems*, 86(2):175–190, Mar 2017. ISSN 1939-8115. doi: 10.1007/s11265-016-1114-9. URL <https://doi.org/10.1007/s11265-016-1114-9>.
- [8] O. Alpar. Frequency spectrograms for biometric keystroke authentication using neural network based classifier. *Knowledge-Based Systems*, 116:163 – 171, 2017. ISSN 0950-7051. doi: <https://doi.org/10.1016/j.knosys.2016.11.006>. URL <http://www.sciencedirect.com/science/article/pii/S0950705116304439>.
- [9] A. Alsultan, K. Warwick, and H. Wei. Non-conventional keystroke dynamics for user authentication. 2017. *Pattern Recognition Letters*.
- [10] A. Alzubaidi and J. Kalita. Authentication of smartphone users using behavioral biometrics. *IEEE Communications Surveys Tutorials*, 18(3):1998–2026, thirdquarter 2016. ISSN 1553-877X. doi: 10.1109/COMST.2016.2537748.
- [11] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz. Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In *Proceedings of the 4th International Conference on Ambient Assisted Living and Home Care, IWAAL'12*, pages 216–223, Berlin, Heidelberg, 2012. Springer-Verlag. ISBN 978-3-642-35394-9. doi: 10.1007/978-3-642-35395-6\_30.

- [12] M. Antal, L. Z. Szabó, and I. László. Keystroke dynamics on android platform. *Procedia Technology*, 19:820–826, 2015.
- [13] T. Anusas-amornkul and K. Wangsuk. A comparison of keystroke dynamics techniques for user authentication. In *2015 International Computer Science and Engineering Conference (ICSEC)*, pages 1–5, Nov 2015. doi: 10.1109/ICSEC.2015.7401401.
- [14] L. C. F. Araujo, L. H. R. Sucupira, M. G. Lizarraga, L. L. Ling, and J. B. T. Yabu-Uti. User authentication through typing biometrics features. *IEEE Transactions on Signal Processing*, 53(2):851–855, Feb 2005. ISSN 1053-587X. doi: 10.1109/TSP.2004.839903.
- [15] P. Arora, M. Hanmandlu, and S. Srivastava. Gait based authentication using gait information image features. *Pattern Recognition Letters*, 68:336–342, 2015.
- [16] D. Asonov and R. Agrawal. Keyboard acoustic emanations. In *2004 IEEE Symposium on Security and Privacy (S&P 2004), 9-12 May 2004, Berkeley, CA, USA*, pages 3–11, 2004. doi: 10.1109/SECPRI.2004.1301311.
- [17] M. M. Association. Midi tuning specification. <http://www.midi.org/techspecs/midituning.php>, [Accessed : 14-March-2019].
- [18] G. L. Azevedo, G. D. Cavalcanti, and E. C. Carvalho Filho. An approach to feature selection for keystroke dynamics systems based on pso and feature weighting. In *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*, pages 3577–3584. IEEE, 2007.
- [19] K. Balagani, V. Phoha, A. Ray, and S. Phoha. On the discriminability of keystroke feature vectors used in fixed text keystroke authentication. 2011. *Pattern Recognition Letters*.
- [20] K. S. Balagani, V. V. Phoha, A. Ray, and S. Phoha. On the discriminability of keystroke feature vectors used in fixed text keystroke authentication. *Pattern Recognition Letters*, 32(7):1070 – 1080, 2011. ISSN 0167-8655. doi: <https://doi.org/10.1016/j.patrec.2011.02.014>. URL <http://www.sciencedirect.com/science/article/pii/S0167865511000511>.
- [21] R. Banerjee, S. Feng, J. S. Kang, and Y. Choi. Keystroke patterns as prosody in digital writings: A case study with deceptive reviews and essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1469–1473, Doha, Qatar, October 2014. Association for Computational Linguistics. URL <http://www.aclweb.org/anthology/D14-1155>.
- [22] P. Barford and M. Crovella. Generating representative web workloads for network and server performance evaluation. *ACM SIGMETRICS Performance Evaluation Review*, 26(1):151–160, 1998.
- [23] A. K. Belman and V. V. Phoha. Doubletype: Authentication using relationship between typing behavior on multiple devices. In *2020 International Conference on Artificial Intelligence and Signal Processing (AISP)*, pages 1–6, 2020.
- [24] A. K. Belman and V. V. Phoha. Discriminative power of typing features on desktops, tablets, and phones for user identification. *ACM Trans. Priv. Secur.*, 23(1), Feb. 2020. ISSN 2471-2566. doi: 10.1145/3377404. URL <https://doi.org/10.1145/3377404>.

- [25] A. K. Belman, L. Wang, S. S. Iyengar, P. Sniatala, R. Wright, R. Dora, J. Baldwin, Z. Jin, and V. V. Phoha. Insights from bb-mas – a large dataset for typing, gait and swipes of the same person on desktop, tablet and phone, 2019. URL <https://arxiv.org/abs/1912.02736>.
- [26] A. K. Belman, L. Wang, S. S. Iyengar, P. Sniatala, R. Wright, R. Dora, J. Baldwin, Z. Jin, and V. V. Phoha. Su-ais bb-mas (syracuse university and assured information security - behavioral biometrics multi-device and multi-activity data from same users) dataset, 2019. URL <http://dx.doi.org/10.21227/rpaz-0h66>.
- [27] A. K. Belman, T. Paul, L. Wang, S. S. Iyengar, P. Śniatała, Z. Jin, V. V. Phoha, S. Vainio, and J. Roning. Authentication by mapping keystrokes to music: The melody of typing. In *2020 International Conference on Artificial Intelligence and Signal Processing (AISP)*, pages 1–6, 2020.
- [28] A. K. Belman, S. Sridhara, and V. V. Phoha. Classification of threat level in typing activity through keystroke dynamics. In *2020 International Conference on Artificial Intelligence and Signal Processing (AISP)*, pages 1–6, 2020.
- [29] F. Bergadano, D. Gunetti, and C. Picardi. User authentication through keystroke dynamics. *ACM Trans. Inf. Syst. Secur.*, 5(4):367–397, Nov. 2002. ISSN 1094-9224. doi: 10.1145/581271.581272.
- [30] A. Bhatia, M. Hanmandlu, S. Vasikarla, and B. K. Panigrahi. Keystroke dynamics based authentication using gfm. In *2018 IEEE International Symposium on Technologies for Homeland Security (HST)*, pages 1–5, 2018.
- [31] E. Biermann, E. Cloete, and L. Venter. A comparison of intrusion detection systems. *Computers and Security*, 20(8):676 – 683, 2001. ISSN 0167-4048. doi: [https://doi.org/10.1016/S0167-4048\(01\)00806-9](https://doi.org/10.1016/S0167-4048(01)00806-9). URL <http://www.sciencedirect.com/science/article/pii/S0167404801008069>.
- [32] J. Blasco, T. M. Chen, J. Tapiador, and P. Peris-Lopez. A survey of wearable biometric recognition systems. *ACM Comput. Surv.*, 49(3):43:1–43:35, Sept. 2016. ISSN 0360-0300. doi: 10.1145/2968215. URL <http://doi.acm.org/10.1145/2968215>.
- [33] S. Bleha, C. Slivinsky, and B. Hussien. Computer-access security systems using keystroke dynamics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(12):1217–1222, Dec 1990. ISSN 0162-8828. doi: 10.1109/34.62613.
- [34] C. Bo, L. Zhang, X.-Y. Li, Q. Huang, and Y. Wang. Silentsense: Silent user identification via touch and movement behavioral biometrics. In *Proceedings of the 19th Annual International Conference on Mobile Computing and #38; Networking, MobiCom '13*, pages 187–190, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1999-7. doi: 10.1145/2500423.2504572.
- [35] D. G. Brizan, A. Goodkind, P. Koch, K. Balagani, V. V. Phoha, and A. Rosenberg. Utilizing linguistically enhanced keystroke dynamics to predict typist cognition and demographics. *International Journal of Human-Computer Studies*, 82:57 – 68, 2015. ISSN 1071-5819. doi: <https://doi.org/10.1016/j.ijhcs.2015.04.005>. URL <http://www.sciencedirect.com/science/article/pii/S107158191500097X>.

- [36] D. G. Brizan, A. Goodkind, P. Koch, K. Balagani, V. V. Phoha, and A. Rosenberg. Utilizing linguistically enhanced keystroke dynamics to predict typist cognition and demographics. *International Journal of Human-Computer Studies*, 82:57 – 68, 2015. ISSN 1071-5819. doi: <https://doi.org/10.1016/j.ijhcs.2015.04.005>.
- [37] S. Caballero-Morales and A. Rahim. Analyzing the effect of non-normality on the solution space for the economic statistical design of x-bar control charts. In *2015 International Conference on Industrial Engineering and Operations Management (IEOM)*, pages 1–6, March 2015. doi: 10.1109/IEOM.2015.7093766.
- [38] S. Caballero-Morales and A. Rahim. Analyzing the effect of non-normality on the solution space for the economic statistical design of x-bar control charts. In *2015 International Conference on Industrial Engineering and Operations Management (IEOM)*, pages 1–6, March 2015. doi: 10.1109/IEOM.2015.7093766.
- [39] J. M. Chang, C. Fang, K. Ho, N. Kelly, P. Wu, Y. Ding, C. Chu, S. Gilbert, A. E. Kamal, and S. Kung. Capturing cognitive fingerprints from keystroke dynamics. *IT Professional*, 15(4):24–28, July 2013. ISSN 1520-9202. doi: 10.1109/MITP.2013.52.
- [40] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *J. Artif. Int. Res.*, 16(1):321–357, June 2002. ISSN 1076-9757. URL <http://dl.acm.org/citation.cfm?id=1622407.1622416>.
- [41] Y. Chen, T. Li, R. Zhang, Y. Zhang, and T. Hedgpeth. Eytell: Video-assisted touchscreen keystroke inference from eye movements. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 144–160, May 2018. doi: 10.1109/SP.2018.00010.
- [42] R. Chereshevnev and A. Kertész-Farkas. Hugadb: Human gait database for activity recognition from wearable inertial sensor networks. *CoRR*, abs/1705.08506, 2017. URL <http://arxiv.org/abs/1705.08506>.
- [43] F. Ciuffo and G. M. Weiss. Smartwatch-based transcription biometrics. In *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pages 145–149, Oct 2017. doi: 10.1109/UEMCON.2017.8249014.
- [44] N. Clarke and S. Furnell. Advanced user authentication for mobile devices. *Computers and Security*, 26(2):109 – 119, 2007. ISSN 0167-4048. doi: <https://doi.org/10.1016/j.cose.2006.08.008>.
- [45] N. Clarke, S. Furnell, B. Lines, and P. Reynolds. Keystroke dynamics on a mobile handset: a feasibility study. *Information Management and Computer Security*, 11(4):161–166, 2003. doi: 10.1108/09685220310489526. URL <https://doi.org/10.1108/09685220310489526>.
- [46] N. Clarke, S. Furnell, B. Lines, and P. Reynolds. Keystroke dynamics on a mobile handset: a feasibility study. *Information Management and Computer Security*, 11(4):161–166, 2003. doi: 10.1108/09685220310489526.
- [47] N. L. Clarke and S. M. Furnell. Authenticating mobile phone users using keystroke analysis. 2006. *Int. J. Inf. Secur.*

- [48] N. L. Clarke and S. M. Furnell. Authenticating mobile phone users using keystroke analysis. *International Journal of Information Security*, 6(1):1–14, Jan 2007. ISSN 1615-5270. doi: 10.1007/s10207-006-0006-6.
- [49] H. Cramér. *Mathematical methods of statistics*. Princeton University Press, Princeton, 1999.
- [50] H. Crawford. Keystroke dynamics: Characteristics and opportunities. In *2010 Eighth International Conference on Privacy, Security and Trust*, pages 205–212, Aug 2010. doi: 10.1109/PST.2010.5593258.
- [51] M. Davies. Corpus of contemporary american english. 2010. URL <https://www.english-corpora.org/coca/>. (Updated 2017, Accessed August 29, 2019).
- [52] H. Davoudi and E. Kabir. A new distance measure for free text keystroke authentication. In *2009 14th International CSI Computer Conference*, pages 570–575, Oct 2009. doi: 10.1109/CSICC.2009.5349640.
- [53] K. Delac and M. Grgic. A survey of biometric recognition methods. In *Proceedings. Elmar-2004. 46th International Symposium on Electronics in Marine*, pages 184–193, June 2004.
- [54] I. Deutschmann, P. Nordström, and L. Nilsson. Continuous authentication using behavioral biometrics. *IT Professional*, 15(4):12–15, July 2013. ISSN 1520-9202. doi: 10.1109/MITP.2013.50.
- [55] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Comput. Linguist.*, 19(1):61–74, Mar. 1993. ISSN 0891-2017.
- [56] C. Epp, M. Lippold, and R. L. Mandryk. Identifying emotional states using keystroke dynamics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 715–724, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0228-9. doi: 10.1145/1978942.1979046. URL <http://doi.acm.org/10.1145/1978942.1979046>.
- [57] S. Fang, I. Markwood, Y. Liu, S. Zhao, Z. Lu, and H. Zhu. No training hurdles: Fast training-agnostic attacks to infer your typing. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 1747–1760. ACM, 2018.
- [58] A. M. Feit, D. Weir, and A. Oulasvirta. How we type: Movement strategies and performance in everyday typing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 4262–4273, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3362-7. doi: 10.1145/2858036.2858233. URL <http://doi.acm.org/10.1145/2858036.2858233>.
- [59] T. Feng, Z. Liu, K. Kwon, W. Shi, B. Carbutar, Y. Jiang, and N. Nguyen. Continuous mobile authentication using touchscreen gestures. In *2012 IEEE Conference on Technologies for Homeland Security (HST)*, pages 451–456, Nov 2012. doi: 10.1109/THS.2012.6459891.
- [60] J. Fierrez, A. Pozo, M. Martinez-Diaz, J. Galbally, and A. Morales. Benchmarking touchscreen biometrics for mobile authentication. *IEEE Transactions on Information Forensics and Security*, 13(11):2720–2733, 2018.

- [61] B. C. Florea. Midi-based controller of electrical drives. In *Proceedings of the 2014 6th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pages 27–30, Oct 2014. doi: 10.1109/ECAI.2014.7090159.
- [62] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. *Trans. Info. For. Sec.*, 8(1):136–148, Jan. 2013. ISSN 1556-6013. doi: 10.1109/TIFS.2012.2225048. URL <http://dx.doi.org/10.1109/TIFS.2012.2225048>.
- [63] C. A. Fukuchi, R. K. Fukuchi, and M. Duarte. A public dataset of overground and treadmill walking kinematics and kinetics in healthy individuals. *PeerJ*, 6:e4640–e4640, Apr 2018. ISSN 2167-8359. doi: 10.7717/peerj.4640. URL <https://www.ncbi.nlm.nih.gov/pubmed/29707431>.
- [64] D. Gafurov, K. Helkala, and T. Søndrol. Biometric gait authentication using accelerometer sensor. *JCP*, 1(7):51–59, 2006.
- [65] D. Gafurov, E. Snekkenes, and P. Bours. Gait authentication and identification using wearable accelerometer sensor. In *2007 IEEE workshop on automatic identification advanced technologies*, pages 220–225. IEEE, 2007.
- [66] Gaines, S. R., W. Lisowski, S. James, and N. Shapiro. Authentication by keystroke timing: Some preliminary results. 1980.
- [67] A. Goodkind, D. G. Brizan, and A. Rosenberg. Utilizing overt and latent linguistic structure to improve keystroke-based authentication. *Image and Vision Computing*, 58:230 – 238, 2017. ISSN 0262-8856. doi: <https://doi.org/10.1016/j.imavis.2016.06.003>. URL <http://www.sciencedirect.com/science/article/pii/S0262885616301019>.
- [68] M. T. Goodrich, M. Sirivianos, J. Solis, G. Tsudik, and E. Uzun. Loud and clear: Human-verifiable authentication based on audio. In *26th IEEE International Conference on Distributed Computing Systems (ICDCS'06)*, pages 10–10, July 2006. doi: 10.1109/ICDCS.2006.52.
- [69] D. Gunetti and C. Picardi. Keystroke analysis of free text. *ACM Trans. Inf. Syst. Secur.*, 8(3):312–347, Aug. 2005. ISSN 1094-9224. doi: 10.1145/1085126.1085129. URL <http://doi.acm.org/10.1145/1085126.1085129>.
- [70] D. Gunetti and C. Picardi. Keystroke analysis of free text. *ACM Trans. Inf. Syst. Secur.*, 8(3):312–347, Aug. 2005. ISSN 1094-9224. doi: 10.1145/1085126.1085129.
- [71] D. Gunetti and C. Picardi. Keystroke analysis of free text. *ACM Trans. Inf. Syst. Secur.*, 8(3):312–347, Aug. 2005. ISSN 1094-9224. doi: 10.1145/1085126.1085129. URL <http://doi.acm.org/10.1145/1085126.1085129>.
- [72] D. Gunetti, C. Picardi, and G. Ruffo. Dealing with different languages and old profiles in keystroke analysis of free text, 09 2005.
- [73] J. Ho and D.-K. Kang. Mini-batch bagging and attribute ranking for accurate user authentication in keystroke dynamics. *Pattern Recognition*, 70:139 – 151, 2017. ISSN 0031-3203. doi: <https://doi.org/10.1016/j.patcog.2017.05.002>. URL <http://www.sciencedirect.com/science/article/pii/S003132031730184X>.

- [74] D. Hosseinzadeh, S. Krishnan, and A. Khademi. Keystroke identification based on gaussian mixture models. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 3, pages III–III, 2006.
- [75] <http://kenschutte.com/midi>. Ken schutte midi matlab toolbox, 2015. <https://github.com/kts/matlab-midi>.
- [76] J. Huang, D. Hou, S. Schuckers, and S. Upadhyaya. Effects of text filtering on authentication performance of keystroke biometrics. 2016. IEEE International Workshop on Information Forensics and Security ( WIFS ).
- [77] J. Huang, D. Hou, and S. Schuckers. A practical evaluation of free-text keystroke dynamics. In *2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*, pages 1–8, Feb 2017. doi: 10.1109/ISBA.2017.7947695.
- [78] R. Janakiraman and T. Sim. Keystroke dynamics in a general setting. In *Proceedings of the 2007 International Conference on Advances in Biometrics, ICB'07*, pages 584–593, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 3-540-74548-3, 978-3-540-74548-8. URL <http://dl.acm.org/citation.cfm?id=2391659.2391726>.
- [79] R. Janakiraman and T. Sim. Keystroke dynamics in a general setting. In S.-W. Lee and S. Z. Li, editors, *Advances in Biometrics*, pages 584–593, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. ISBN 978-3-540-74549-5.
- [80] K. Jin, S. Fang, C. Peng, Z. Teng, X. Mao, L. Zhang, and X. Li. Vivisnoop: Someone is snooping your typing without seeing it! In *2017 IEEE Conference on Communications and Network Security (CNS)*, pages 1–9, Oct 2017. doi: 10.1109/CNS.2017.8228624.
- [81] R. Joyce and G. Gupta. Identity authentication based on keystroke latencies. *Commun. ACM*, 33(2):168–176, Feb. 1990. ISSN 0001-0782. doi: 10.1145/75577.75582.
- [82] R. Joyce and G. Gupta. Identity authentication based on keystroke latencies. *Commun. ACM*, 33(2):168–176, Feb. 1990. ISSN 0001-0782. doi: 10.1145/75577.75582. URL <http://doi.acm.org/10.1145/75577.75582>.
- [83] A. K. Belman, L. Wang, S. S. Iyengar, P. Sniatala, R. Wright, R. Dora, J. Baldwin, Z. Jin, and V. V. Phoha. Su-ais bb-mas (syracuse university and assured information security - behavioral biometrics multi-device and multi-activity data from same users) dataset, 2019. URL <http://dx.doi.org/10.21227/rpaz-0h66>.
- [84] S. Karatzouni and N. Clarke. Keystroke analysis for thumb-based keyboards on mobile devices. In H. Venter, M. Eloff, L. Labuschagne, J. Eloff, and R. von Solms, editors, *New Approaches for Security, Privacy and Trust in Complex Environments*, pages 253–263, Boston, MA, 2007. Springer US. ISBN 978-0-387-72367-9.
- [85] M. Karnan and N. Krishnaraj. Bio password - keystroke dynamic approach to secure mobile devices. In *2010 IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–4, Dec 2010. doi: 10.1109/ICCIC.2010.5705901.



- [86] H. Khan, U. Hengartner, and D. Vogel. Augmented reality-based mimicry attacks on behaviour-based smartphone authentication. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys '18*, pages 41–53, New York, NY, USA, 2018. ACM. ISBN 978-1-4503-5720-3. doi: 10.1145/3210240.3210317. URL <http://doi.acm.org/10.1145/3210240.3210317>.
- [87] K. S. Killourhy and R. A. Maxion. Comparing anomaly-detection algorithms for keystroke dynamics. In *2009 IEEE/IFIP International Conference on Dependable Systems & Networks*, pages 125–134. IEEE, 2009.
- [88] K. S. Killourhy and R. A. Maxion. Free vs. transcribed text for keystroke-dynamics evaluations. In *Proceedings of the 2012 Workshop on Learning from Authoritative Security Experiment Results*, pages 1–8. ACM, 2012.
- [89] E. M. Kim. Harmonograph: A visual representation of sound. In *2012 International Symposium on Intelligent Signal Processing and Communications Systems*, pages 489–494, Nov 2012. doi: 10.1109/ISPACS.2012.6473539.
- [90] J. Kim, H. Kim, and P. Kang. Keystroke dynamics-based user authentication using freely typed text based on user-adaptive feature extraction and novelty detection. *Applied Soft Computing*, 62:1077 – 1087, 2018. ISSN 1568-4946. doi: <https://doi.org/10.1016/j.asoc.2017.09.045>.
- [91] S. G. Kwak and J. H. Kim. Central limit theorem: the cornerstone of modern statistics. *Korean journal of anesthesiology*, 70(2):144–156, Apr 2017. ISSN 2005-6419. doi: 10.4097/kjae.2017.70.2.144. URL <https://pubmed.ncbi.nlm.nih.gov/28367284>. 28367284[pmid].
- [92] A. C. M. Kwan. Validity of normality assumption in csp research. In N. Foo and R. Goebel, editors, *PRICAI'96: Topics in Artificial Intelligence*, pages 253–263, Berlin, Heidelberg, 1996. Springer Berlin Heidelberg. ISBN 978-3-540-68729-0.
- [93] K.-C. Lan and J. Heidemann. Rapid model parameterization from traffic measurements. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 12(3):201–229, 2002.
- [94] H. S. Lee, T. S. Lau, W. K. Lai, Y. C. King, and L. L. Lim. User identification of numerical keypad typing patterns with subtractive clustering fuzzy inference. In *2017 IEEE 15th Student Conference on Research and Development (SCORED)*, pages 83–88, Dec 2017. doi: 10.1109/SCORED.2017.8305416.
- [95] J. Leggett and G. Williams. Verifying identity via keystroke characteristics. *International Journal of Man-Machine Studies*, 28(1):67 – 76, 1988. ISSN 0020-7373. doi: [https://doi.org/10.1016/S0020-7373\(88\)80053-1](https://doi.org/10.1016/S0020-7373(88)80053-1).
- [96] E. L. Lehmann and G. Casella. *Theory of point estimation / E.L. Lehmann, George Casella*. Springer, 1998.
- [97] B. Li, H. Sun, Y. Gao, V. V. Phoha, and Z. Jin. Enhanced free-text keystroke continuous authentication based on dynamics of wrist motion. In *2017 IEEE Workshop on Information Forensics and Security, WIFS 2017, Rennes, France, December 4-7, 2017*, pages 1–6, 2017. doi: 10.1109/WIFS.2017.8267642. URL <https://doi.org/10.1109/WIFS.2017.8267642>.

- [98] L. Li, X. Zhao, and G. Xue. Unobservable re-authentication for smart-phones. In *20th Annual Network and Distributed System Security Symposium, NDSS 2013, San Diego, California, USA, February 24-27, 2013*, 2013. URL <https://www.ndss-symposium.org/ndss2013/unobservable-re-authentication-smartphones>.
- [99] X. Li, J. Le, P. Gopalakrishnan, and L. T. Pileggi. Asymptotic probability extraction for nonnormal performance distributions. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 26(1):16–37, 2006.
- [100] H. W. Lilliefors. On the kolmogorov-smirnov test for normality with mean and variance unknown. *Journal of the American statistical Association*, 62(318):399–402, 1967.
- [101] E. Limpert and W. A. Stahel. Problems with using the normal distribution—and ways to improve quality and efficiency of data analysis. *PloS one*, 6(7):e21403–e21403, 2011. ISSN 1932-6203. doi: 10.1371/journal.pone.0021403. URL <https://www.ncbi.nlm.nih.gov/pubmed/21779325>. 21779325[pmid].
- [102] E. Limpert and W. A. Stahel. Problems with using the normal distribution – and ways to improve quality and efficiency of data analysis. *PLOS ONE*, 6(7):1–8, 07 2011. doi: 10.1371/journal.pone.0021403. URL <https://doi.org/10.1371/journal.pone.0021403>.
- [103] J. Liu, Y. Wang, G. Kar, Y. Chen, J. Yang, and M. Gruteser. Snooping keystrokes with mm-level audio ranging on a single phone. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, MobiCom ’15*, pages 142–154, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3619-2. doi: 10.1145/2789168.2790122. URL <http://doi.acm.org/10.1145/2789168.2790122>.
- [104] J. Liu, Y. Wang, G. Kar, Y. Chen, J. Yang, and M. Gruteser. Snooping keystrokes with mm-level audio ranging on a single phone. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, MobiCom ’15*, pages 142–154, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3619-2. doi: 10.1145/2789168.2790122. URL <http://doi.acm.org/10.1145/2789168.2790122>.
- [105] J. Liu, Z. Wu, J. Wu, J. Dong, Y. Zhao, and D. Wen. A weibull distribution accrual failure detector for cloud computing. *PLOS ONE*, 12(3):1–16, 03 2017. doi: 10.1371/journal.pone.0173666. URL <https://doi.org/10.1371/journal.pone.0173666>.
- [106] C. C. Loy, C. P. Lim, and W. K. Lai. Pressure-based typing biometrics user authentication using the fuzzy artmap neural network. 2005.
- [107] L. Lu, J. Yu, Y. Chen, Y. Zhu, X. Xu, G. Xue, and M. Li. Keyliesterber: Inferring keystrokes on qwerty keyboard of touch screen through acoustic signals. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pages 775–783, April 2019. doi: 10.1109/INFOCOM.2019.8737591.
- [108] E. Maiorana, P. Campisi, N. González-Carballo, and A. Neri. Keystroke dynamics authentication for mobile phones. In *Proceedings of the 2011 ACM Symposium on Applied Computing, SAC ’11*, pages 21–26, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0113-8. doi: 10.1145/1982185.1982190.

- [109] A. Maiti, M. Jadliwala, J. He, and I. Bilogrevic. Side-channel inference attacks on mobile keypads using smartwatches. *IEEE Transactions on Mobile Computing*, 17(9):2180–2194, Sep. 2018. ISSN 1536-1233. doi: 10.1109/TMC.2018.2794984.
- [110] N. F. Marko and R. J. Weil. Non-gaussian distributions affect identification of expression patterns, functional annotation, and prospective classification in human cancer genomes. *PLOS ONE*, 7(10):1–15, 10 2012. doi: 10.1371/journal.pone.0046935. URL <https://doi.org/10.1371/journal.pone.0046935>.
- [111] F. J. Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951.
- [112] S. Matta, H. Rudolph, and D. K. Kumar. Auditory eyes: Representing visual information in sound and tactile cues. In *2005 13th European Signal Processing Conference*, pages 1–4, Sep. 2005.
- [113] P. B. L. Meijer. An experimental system for auditory image representations. *IEEE Transactions on Biomedical Engineering*, 39(2):112–121, Feb 1992. ISSN 0018-9294. doi: 10.1109/10.121642.
- [114] D. Micucci, M. Mobilio, and P. Napoletano. Unimib shar: A dataset for human activity recognition using acceleration data from smartphones. *Applied Sciences*, 7(10), 2017. ISSN 2076-3417. doi: 10.3390/app7101101. URL <http://www.mdpi.com/2076-3417/7/10/1101>.
- [115] M. Monaro, C. Galante, R. Spolaor, Q. Q. Li, L. Gamberini, M. Conti, and G. Sartori. Covert lie detection using keyboard dynamics. *Scientific Reports*, 8(1):1976, 2018. ISSN 2045-2322. doi: 10.1038/s41598-018-20462-6. URL <https://doi.org/10.1038/s41598-018-20462-6>.
- [116] S. Mondal and P. Bours. Swipe gesture based continuous authentication for mobile devices. In *2015 International Conference on Biometrics (ICB)*, pages 458–465, May 2015. doi: 10.1109/ICB.2015.7139110.
- [117] S. Mondal and P. Bours. A study on continuous authentication using a combination of keystroke and mouse biometrics. *Neurocomputing*, 230:1–22, 2017. ISSN 0925-2312. doi: <https://doi.org/10.1016/j.neucom.2016.11.031>. URL <http://www.sciencedirect.com/science/article/pii/S0925231216314321>.
- [118] S. Mondal and P. Bours. Person identification by keystroke dynamics using pairwise user coupling. *IEEE Transactions on Information Forensics and Security*, 12(6):1319–1329, June 2017. ISSN 1556-6013. doi: 10.1109/TIFS.2017.2658539.
- [119] F. Monrose and A. Rubin. Authentication via keystroke dynamics. In *Proceedings of the 4th ACM Conference on Computer and Communications Security, CCS '97*, pages 48–56, New York, NY, USA, 1997. ACM. ISBN 0-89791-912-2. doi: 10.1145/266420.266434. URL <http://doi.acm.org/10.1145/266420.266434>.
- [120] F. Monrose and A. D. Rubin. Keystroke dynamics as a biometric for authentication. *Future Generation Computer Systems*, 16(4):351–359, 2000. ISSN 0167-739X. doi: [https://doi.org/10.1016/S0167-739X\(99\)00059-X](https://doi.org/10.1016/S0167-739X(99)00059-X).
- [121] F. Monrose and A. D. Rubin. Keystroke dynamics as a biometric for authentication. *Future Generation computer systems*, 16(4):351–359, 2000.

- [122] F. Monrose and A. D. Rubin. Keystroke dynamics as a biometric for authentication. *Future Generation computer systems*, 16(4):351–359, 2000.
- [123] D. Montgomery and G. Runger. *Applied Statistics and Probability for Engineers, 6th Edition*. John Wiley & Sons, 2013.
- [124] C. Murphy, J. Huang, D. Hou, and S. Schuckers. Shared dataset on natural human-computer interaction to support continuous authentication research. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, pages 525–530, Oct 2017. doi: 10.1109/BTAS.2017.8272738.
- [125] A. Nazmul Haque Nahin, J. Mohammad Alam, H. Mahmud, and K. Hasan. Identifying emotion by keystroke dynamics and text pattern analysis. *Behaviour and Information Technology*, 33, 09 2014. doi: 10.1080/0144929X.2014.907343.
- [126] T. Nguyen and J. Voris. Touchscreen biometrics across multiple devices. 2017. Symposium on Usable Privacy and Security, USENIX.
- [127] T. N. Nguyen, H. H. Huynh, and J. Meunier. 3d reconstruction with time-of-flight depth camera and multiple mirrors. *IEEE Access*, 6:38106–38114, 2018. ISSN 2169-3536. doi: 10.1109/ACCESS.2018.2854262.
- [128] R. Noorossana, A. Vaghefi, and M. Dorri. The effect of non-normality on performance of linear profile monitoring. In *2008 IEEE International Conference on Industrial Engineering and Engineering Management*, pages 262–266, Dec 2008. doi: 10.1109/IEEM.2008.4737871.
- [129] M. Obaidat. A methodology for improving computer access security. *Computers and Security*, 12(7):657 – 662, 1993. ISSN 0167-4048. doi: [https://doi.org/10.1016/0167-4048\(93\)90083-H](https://doi.org/10.1016/0167-4048(93)90083-H).
- [130] M. Obaidat and B. Sadoun. Verification of computer users using keystroke dynamics. 1997. *IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics*. Vol. 27, NO. 2.
- [131] M. S. Obaidat and B. Sadoun. Verification of computer users using keystroke dynamics. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 27(2):261–269, April 1997. ISSN 1083-4419. doi: 10.1109/3477.558812.
- [132] T. I. of Scientific and J. Industrial Research (ISIR), Osaka University (OU). Ou-isir biometric database, 2015. <http://www.am.sanken.osaka-u.ac.jp/BiometricDB/InertialGait.html>, [Accessed : 22-June-2020].
- [133] C. E. Olson Jr. The fallacy of normality in remotely sensed data. In *Proceedings of the ASPRS Annual Conference*, pages 9–13, 2009.
- [134] G. Pahuja and T. N. Nagabhushan. Biometric authentication amp; identification through behavioral biometrics: A survey. In *2015 International Conference on Cognitive Computing and Information Processing(CCIP)*, pages 1–7, March 2015. doi: 10.1109/CCIP.2015.7100681.
- [135] S. Park, E. Serpedin, and K. Qaraqe. Gaussian assumption: The least favorable but the most useful [lecture notes]. *IEEE Signal Processing Magazine*, 30(3):183–186, May 2013. ISSN 1558-0792. doi: 10.1109/MSP.2013.2238691.

- [136] T. Paul, S. Vainio, and J. Roning. Towards personalised, dna signature derived music via the short tandem repeats (str): Proceedings of the 2018 computing conference, volume 2. In *Intelligent Computing. SAI 2018. Advances in Intelligent Systems and Computing, vol 857*. Springer, Cham., pages 951–964, 01 2019. ISBN 978-3-030-01176-5. doi: 10.1007/978-3-030-01177-2\_69.
- [137] J. Pek, O. Wong, and A. C. M. Wong. How to Address Non-normality: A Taxonomy of Approaches, Reviewed, and Illustrated. *Frontiers in Psychology*, 9:2104, Nov. 2018. ISSN 1664-1078. doi: 10.3389/fpsyg.2018.02104.
- [138] A. Pentel. Predicting user age by keystroke dynamics. In R. Silhavy, editor, *Artificial Intelligence and Algorithms in Intelligent Systems*, pages 336–343, Cham, 2019. Springer International Publishing. ISBN 978-3-319-91189-2.
- [139] P. H. Pisani and A. C. Lorena. A systematic review on keystroke dynamics. *Journal of the Brazilian Computer Society*, 19(4):573–587, Nov 2013. ISSN 1678-4804. doi: 10.1007/s13173-013-0117-7. URL <https://doi.org/10.1007/s13173-013-0117-7>.
- [140] A. Pozo, J. Fierrez, M. Martinez-Diaz, J. Galbally, and A. Morales. Exploring a statistical method for touchscreen swipe biometrics. In *2017 International Carnahan Conference on Security Technology (ICCST)*, pages 1–4, 2017.
- [141] O. U. Press. The oxford english corpus: Facts about the language. [oxforddictionaries.com](https://web.archive.org/web/20111226085859/http://oxforddictionaries.com/words/the-oec-facts-about-the-language). 2011. URL <https://web.archive.org/web/20111226085859/http://oxforddictionaries.com/words/the-oec-facts-about-the-language>. (Accessed August 29, 2019).
- [142] K. A. Rahman, K. S. Balagani, and V. V. Phoha. Snoop-forge-replay attacks on continuous verification with keystrokes. *IEEE Transactions on Information Forensics and Security*, 8(3):528–541, March 2013. ISSN 1556-6013. doi: 10.1109/TIFS.2013.2244091.
- [143] K. Revett, S. T. de Magalhães, and H. M. D. Santos. Enhancing login security through the use of keystroke input dynamics. In D. Zhang and A. K. Jain, editors, *Advances in Biometrics*, pages 661–667, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. ISBN 978-3-540-31621-3.
- [144] D. I. Rigas and D. Memery. Utilising audio-visual stimuli in interactive information systems: a two domain investigation on auditory metaphors. In *Proceedings. International Conference on Information Technology: Coding and Computing*, pages 190–195, April 2002. doi: 10.1109/ITCC.2002.1000385.
- [145] R. N. Rodrigues, G. F. G. Yared, C. R. do N. Costa, J. B. T. Yabu-Uti, F. Violaro, and L. L. Ling. Biometric access control through numerical keyboards based on keystroke dynamics. In D. Zhang and A. K. Jain, editors, *Advances in Biometrics*, pages 640–646, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. ISBN 978-3-540-31621-3.
- [146] J. Roth, X. Liu, A. Ross, and D. Metaxas. Biometric authentication via keystroke sound. In *2013 International Conference on Biometrics (ICB)*, pages 1–8, June 2013. doi: 10.1109/ICB.2013.6613015.
- [147] J. Roth, X. Liu, A. Ross, and D. Metaxas. Biometric authentication via keystroke sound. In *2013 International Conference on Biometrics (ICB)*, pages 1–8, June 2013. doi: 10.1109/ICB.2013.6613015.

- [148] J. Roth, X. Liu, A. Ross, and D. Metaxas. Investigating the discriminative power of keystroke sound. *IEEE Transactions on Information Forensics and Security*, 10(2): 333–345, Feb 2015. ISSN 1556-6013. doi: 10.1109/TIFS.2014.2374424.
- [149] J. Roth, X. Liu, A. Ross, and D. Metaxas. Investigating the discriminative power of keystroke sound. *IEEE Transactions on Information Forensics and Security*, 10(2): 333–345, Feb 2015. doi: 10.1109/TIFS.2014.2374424.
- [150] V. S and M. Sylviaa S. Intrusion detection system - a study. *International Journal of Security, Privacy and Trust Management*, 4:31–44, 02 2015. doi: 10.5121/ijsp.2015.4104.
- [151] S. Sarkar, P. J. Phillips, Z. Liu, I. R. Vega, P. Grother, and K. W. Bowyer. The humanid gait challenge problem: data sets, performance, and analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(2):162–177, Feb 2005. doi: 10.1109/TPAMI.2005.39.
- [152] N. Saxena and J. H. Watt. Authentication technologies for the blind or visually impaired. In *Proceedings of the 4th USENIX Conference on Hot Topics in Security, HotSec'09*, pages 7–7, Berkeley, CA, USA, 2009. USENIX Association. URL <http://dl.acm.org/citation.cfm?id=1855628.1855635>.
- [153] S. seob Hwang, S. Cho, and S. Park. Keystroke dynamics-based authentication for mobile devices. *Computers & Security*, 28(1):85 – 93, 2009. ISSN 0167-4048. doi: <https://doi.org/10.1016/j.cose.2008.10.002>. URL <http://www.sciencedirect.com/science/article/pii/S0167404808000965>.
- [154] A. Serwadda and V. V. Phoha. Examining a large keystroke biometrics dataset for statistical-attack openings. *ACM Trans. Inf. Syst. Secur.*, 16(2):8:1–8:30, Sept. 2013. ISSN 1094-9224. doi: 10.1145/2516960. URL <http://doi.acm.org/10.1145/2516960>.
- [155] A. Serwadda and V. V. Phoha. Examining a large keystroke biometrics dataset for statistical-attack openings. *ACM Trans. Inf. Syst. Secur.*, 16(2):8:1–8:30, Sept. 2013. ISSN 1094-9224. doi: 10.1145/2516960. URL <http://doi.acm.org/10.1145/2516960>.
- [156] A. Serwadda, V. V. Phoha, and A. Kiremire. Using global knowledge of users' typing traits to attack keystroke biometrics templates. In *Proceedings of the Thirteenth ACM Multimedia Workshop on Multimedia and Security, MM&#38;Sec '11*, pages 51–60, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0806-9. doi: 10.1145/2037252.2037263. URL <http://doi.acm.org/10.1145/2037252.2037263>.
- [157] A. Serwadda, V. V. Phoha, and Z. Wang. Which verifiers work?: A benchmark evaluation of touch-based authentication algorithms. In *IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems, BTAS 2013, Arlington, VA, USA, September 29 - October 2, 2013*, pages 1–8, 2013. doi: 10.1109/BTAS.2013.6712758. URL <https://doi.org/10.1109/BTAS.2013.6712758>.
- [158] A. Serwadda, Z. Wang, P. Koch, S. Govindarajan, R. Pokala, A. Goodkind, D. Brizan, A. Rosenberg, V. V. Phoha, and K. Balagani. Scan-based evaluation of continuous keystroke authentication systems. *IT Professional*, 15(4):20–23, July 2013. ISSN 1520-9202. doi: 10.1109/MITP.2013.51.

- [159] S. S. SHAPIRO and M. B. WILK. An analysis of variance test for normality. *Biometrika*, 52(3-4):591–611, 12 1965. ISSN 0006-3444. doi: 10.1093/biomet/52.3-4.591. URL <https://doi.org/10.1093/biomet/52.3-4.591>.
- [160] Y. Sheng, V. Phoha, and S. Rovnyak. A parallel decision tree-based method for user authentication based on keystroke patterns. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 35:826 – 833, 09 2005. doi: 10.1109/TSMCB.2005.846648.
- [161] D. Shukla and V. V. Phoha. Stealing passwords by observing hands movement. *IEEE Transactions on Information Forensics and Security*, 14(12):3086–3101, Dec 2019. ISSN 1556-6013. doi: 10.1109/TIFS.2019.2911171.
- [162] A. Siami Namin, R. Hewett, K. S. Jones, and R. Pogrund. Sonifying internet security threats. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, CHI EA '16*, pages 2306–2313, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4082-3. doi: 10.1145/2851581.2892363. URL <http://doi.acm.org/10.1145/2851581.2892363>.
- [163] T. Sim and R. Janakiraman. Are digraphs good for free-text keystroke dynamics? In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6, June 2007. doi: 10.1109/CVPR.2007.383393.
- [164] D. X. Song, D. Wagner, and X. Tian. Timing analysis of keystrokes and timing attacks on ssh. In *Proceedings of the 10th Conference on USENIX Security Symposium - Volume 10, SSYM'01*, Berkeley, CA, USA, 2001. USENIX Association. URL <http://dl.acm.org/citation.cfm?id=1251327.1251352>.
- [165] D. Soria, J. M. Garibaldi, F. Ambrogi, E. M. Biganzoli, and I. O. Ellis. A "non-parametric" version of the naive bayes classifier. *Knowledge-Based Systems*, 24(6):775–784, August 2011. doi: doi:10.1016/j.knosys.2011.02.014. URL <http://eprints.nottingham.ac.uk/28135/>.
- [166] D. Stefan and D. Yao. Keystroke-dynamics authentication against synthetic forgeries. In *6th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2010)*, pages 1–8, Oct 2010. doi: 10.4108/icst.collaboratecom.2010.16.
- [167] D. Stefan, X. Shu, and D. D. Yao. Robustness of keystroke-dynamics based biometrics against synthetic forgeries. *Computers & Security*, 31(1):109 – 121, 2012. ISSN 0167-4048. doi: <https://doi.org/10.1016/j.cose.2011.10.001>.
- [168] M. Stephens. Edf statistics for goodness-of-fit: Part 1. Technical report, STANFORD UNIV CA DEPT OF STATISTICS, 1972.
- [169] A. Stisen, H. Blunck, S. Bhattacharya, T. S. Prentow, M. B. Kjærgaard, A. Dey, T. Sonne, and M. M. Jensen. Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems, SenSys '15*, pages 127–140, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3631-4. doi: 10.1145/2809695.2809718. URL <http://doi.acm.org/10.1145/2809695.2809718>.
- [170] P. Stoica and P. Babu. The gaussian data assumption leads to the largest cramér-rao bound [lecture notes]. *IEEE Signal Processing Magazine*, 28(3):132–133, May 2011. ISSN 1558-0792. doi: 10.1109/MSP.2011.940411.

- [171] A. E. Sulavko, A. V. Eremenko, and A. A. Fedotov. Users' identification through keystroke dynamics based on vibration parameters and keyboard pressure. In *2017 Dynamics of Systems, Mechanisms and Machines (Dynamics)*, pages 1–7, Nov 2017. doi: 10.1109/Dynamics.2017.8239514.
- [172] A. Sulong, Wahyudi, and M. U. Siddiqi. Intelligent keystroke pressure-based typing biometrics authentication system using radial basis function network. In *2009 5th International Colloquium on Signal Processing Its Applications*, pages 151–155, March 2009. doi: 10.1109/CSPA.2009.5069206.
- [173] Y. Sun, H. Ceker, and S. Upadhyaya. Shared keystroke dataset for continuous authentication. In *2016 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, Dec 2016. doi: 10.1109/WIFS.2016.7823894.
- [174] Y. Sun, H. Ceker, and S. Upadhyaya. Anatomy of secondary features in keystroke dynamics - achieving more with less. In *2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*, pages 1–6, Feb 2017. doi: 10.1109/ISBA.2017.7947691.
- [175] C. Tappert and M. Villani. *Keystroke biometric identification studies on long-text input*. PhD thesis, Pace University, USA, 2007.
- [176] P. S. Teh, A. B. J. Teoh, and S. Yue. A survey of keystroke dynamics biometrics. *The Scientific World Journal*, 2013, Sept. 2013. ISSN 2356-6140. doi: 10.1155/2013/408280.
- [177] I. Tsimperidis, A. Arampatzis, and A. Karakos. Keystroke dynamics features for gender recognition. *Digital Investigation*, 24:4 – 10, 2018. ISSN 1742-2876. doi: <https://doi.org/10.1016/j.diin.2018.01.018>. URL <http://www.sciencedirect.com/science/article/pii/S174228761730364X>.
- [178] M. Ulinskas, R. Damaševičius, R. Maskeliūnas, and M. Woźniak. Recognition of human daytime fatigue using keystroke data. *Procedia Computer Science*, 130:947 – 952, 2018. ISSN 1877-0509. doi: <https://doi.org/10.1016/j.procs.2018.04.094>. URL <http://www.sciencedirect.com/science/article/pii/S1877050918304563>. The 9th International Conference on Ambient Systems, Networks and Technologies (ANT 2018) / The 8th International Conference on Sustainable Energy Information Technology (SEIT-2018) / Affiliated Workshops.
- [179] Umphress and G. Williams. Identity verification through keyboard characteristics, 1985. *Int. J. Man-Mach. Stud.*, pp. 263-273.
- [180] Umphress and G. Williams. Identity verification through keyboard characteristics, 1985. *Int. J. Man-Mach. Stud.*, pp. 263-273.
- [181] V. M. Volkova. Research of tukey's test statistic distribution under failure of the normality assumption. In *2016 13th International Scientific-Technical Conference on Actual Problems of Electronics Instrument Engineering (APEIE)*, volume 02, pages 227–229, Oct 2016. doi: 10.1109/APEIE.2016.7806456.
- [182] E. Vural, J. Huang, D. Hou, and S. Schuckers. Shared research dataset to support development of keystroke authentication. In *IEEE International Joint Conference on Biometrics*, pages 1–8. IEEE.
- [183] C. Wan, L. Wang, and V. V. Phoha. A survey on gait recognition. *ACM Comput. Surv.*, 51(5):89:1–89:35, 2019. URL <https://dl.acm.org/citation.cfm?id=3230633>.



- [184] C. Warrender, S. Forrest, and B. Pearlmutter. Detecting intrusions using system calls: alternative data models. In *Proceedings of the 1999 IEEE Symposium on Security and Privacy (Cat. No.99CB36344)*, pages 133–145, May 1999. doi: 10.1109/SECPRI.1999.766910.
- [185] R. Wong, N. Poh, J. Kittler, and D. Frohlich. Towards inclusive design in mobile biometry. In *3rd International Conference on Human System Interaction*, pages 267–274, May 2010. doi: 10.1109/HSI.2010.5514556.
- [186] C. Wu, W. Ding, R. Liu, J. Wang, A. C. Wang, J. Wang, S. Li, Y. Zi, and Z. L. Wang. Keystroke dynamics enabled authentication and identification using triboelectric nanogenerator array. *Materials Today*, 21(3):216 – 222, 2018. ISSN 1369-7021. doi: <https://doi.org/10.1016/j.mattod.2018.01.006>. URL <http://www.sciencedirect.com/science/article/pii/S1369702117307642>.
- [187] [www.cde.ca.gov](http://www.cde.ca.gov). Visual and performing arts framework, 2004. <http://www.cde.ca.gov/ci/cr/cf/documents/vpaframewrk.pdf>, [Accessed : 14-March-2019].
- [188] N. Yager and T. Dunstone. The biometric menagerie. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(2):220–230, Feb 2010. doi: 10.1109/TPAMI.2008.291.
- [189] R. V. Yampolskiy. Indirect human computer interaction-based biometrics for intrusion detection systems. In *2007 41st Annual IEEE International Carrihan Conference on Security Technology*, pages 138–145, Oct 2007. doi: 10.1109/CCST.2007.4373481.
- [190] J. Yang, Y. Li, and M. Xie. Motionauth: Motion-based authentication for wrist worn smart devices. In *2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, pages 550–555. IEEE, 2015.
- [191] Y. Yang and H. Zhu. A study of non-normal process capability analysis based on box-cox transformation. In *2018 3rd International Conference on Computational Intelligence and Applications (ICCI)*, pages 240–243, July 2018. doi: 10.1109/ICCI.2018.00053.
- [192] Y. Yang, D. Li, and Y. Qi. An approach to non-normal process capability analysis using johnson transformation. In *2018 IEEE 4th International Conference on Control Science and Systems Engineering (ICCSSE)*, pages 495–498, Aug 2018. doi: 10.1109/CCSSE.2018.8724679.
- [193] S. Yu, D. Tan, and T. Tan. A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition. In *Proceedings of the 18th International Conference on Pattern Recognition - Volume 04, ICPR '06*, pages 441–444, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2521-0. doi: 10.1109/ICPR.2006.67. URL <https://doi.org/10.1109/ICPR.2006.67>.
- [194] G. Zamonsky Pedernera, S. Sznur, G. Sorondo Ovando, S. García, and G. Meschino. Revisiting clustering methods to their application on keystroke dynamics for intruder classification. In *2010 IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications*, pages 36–40, Sep. 2010. doi: 10.1109/BIOMS.2010.5610443.

- [195] T. Zhu, Q. Ma, S. Zhang, and Y. Liu. Context-free attacks using keyboard acoustic emanations. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, CCS '14*, pages 453–464, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2957-6. doi: 10.1145/2660267.2660296. URL <http://doi.acm.org/10.1145/2660267.2660296>.
- [196] T. Zhu, Q. Ma, S. Zhang, and Y. Liu. Context-free attacks using keyboard acoustic emanations. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, CCS '14*, pages 453–464, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2957-6. doi: 10.1145/2660267.2660296. URL <http://doi.acm.org/10.1145/2660267.2660296>.
- [197] L. Zhuang, F. Zhou, and J. D. Tygar. Keyboard acoustic emanations revisited. *ACM Trans. Inf. Syst. Secur.*, 13(1):3:1–3:26, Nov. 2009. ISSN 1094-9224. doi: 10.1145/1609956.1609959. URL <http://doi.acm.org/10.1145/1609956.1609959>.
- [198] J. Zulueta, P. Andrea, M. Rasic, R. Easter, P. Babu, S. Langenecker, M. McInnis, O. Ajilore, P. Nelson, K. Ryan, and A. Leow. 481. predicting mood disturbance severity in bipolar subjects with mobile phone keystroke dynamics and metadata. *Biological Psychiatry*.
- [199] H. Çeker and S. Upadhyaya. Enhanced recognition of keystroke dynamics using gaussian mixture models. In *MILCOM 2015 - 2015 IEEE Military Communications Conference*, pages 1305–1310, 2015.

VITA

## VITA

### AMITH KAMATH BELMAN

amithbkamath@gmail.com

EECS, Syracuse University

60 Presidential Plaza, Syracuse, NY, 13202

### EDUCATION

---

- **Ph.D. in Computer & Info. Science & Eng, Syracuse University, NY, USA, 2020**
- **Master's of Technology in Computer Science & Eng, Visvesvaraya Tech. University, India, 2013**
- **Bachelor of Engineering in Computer Science, Visvesvaraya Tech. University, India, 2011**

### EXPERIENCE

---

- **Research Assistant** Jun 2017 -  
**Systems Security & Machine Learning Lab, Syracuse University, NY, USA**
- **Graduate Research/ Fellowship** Aug 2015 - May 2017  
**Systems Security & Machine Learning Lab, Syracuse University, NY, USA**
- **Assistant Professor** Jul 2013 - Apr 2015  
**Dept. of Computer Science & Eng., SJB Institute of Technology, Bangalore, India**

### Publications

---

#### *Benchmark Dataset*

- **Amith K. Belman**, Li Wang, Sundaraja S. Iyengar, Pawel Sniatala, Robert Wright, Robert Dora, Jacob Baldwin, Zhanpeng Jin, Vir V. Phoha, "SU-AIS BB-MAS (Syracuse University and Assured Information Security - Behavioral Biometrics Multi-device and multi-Activity data from Same users) Dataset ", IEEE Dataport, 2019.

#### *Journals & Conferences*

- **Amith K. Belman** and Vir V. Phoha, "Discriminative Power of Typing Features on Desktops, Tablets, and Phones for User Identification". In: *ACM Transactions on Privacy and Security*. 23, 1, Article 4 (Feb. 2020), 36 pages.
- **Amith K. Belman**, Swathi Sridhara and Vir Phoha, "Classification of Threat Level in Typing Activity Through Keystroke Dynamics". In: *2020 International Conference on Artificial Intelligence and Signal Processing (AISP)*, 2020.

- **Amith K. Belman**, and Vir Phoha, “DoubleType: Authentication Using Relationship Between Typing Behavior on Multiple Devices”. In: *2020 International Conference on Artificial Intelligence and Signal Processing (AISP)*, 2020.
- **Amith K. Belman**, Tirthankar Paul, Li Wang, S. S. Iyengar, Paweł Śniatała, Zhanpeng Jin, Vir V. Phoha, Seppo Vainio and Juha Röning, “Authentication by Mapping Keystrokes to Music: The Melody of Typing”. In: *2020 International Conference on Artificial Intelligence and Signal Processing (AISP)*, 2020.
- Mingtao Wu, Vir V. Phoha, Young B. Moon, and **Amith K. Belman**, “Detecting Malicious Defects in 3D Printing Process Using Machine Learning and Image Classification”. In: *Proceedings of the ASME 2016 International Mechanical Engineering Congress and Exposition*, Volume 14. Phoenix, Arizona, USA. (Nov. 2016).
- J. Majumdar, G. M. Venkatesh, **Amith K. Belman**, “Video Shot detection using Corner detectors and Optical flow”. In: *International Conference on Emerging Research in Computing, Information, Communication and Applications (ERCICA-2013)*, (Aug. 2013).  
*Accepted/In first look evaluation*
- Li Wang, Sitharama Iyengar, **Amith K. Belman**, Paweł Śniatała, Vir V Phoha, Changsheng Wan “Game Theory based Cyber Insurance to cover Potential Loss from Mobile Malware Exploitation”. In: *ACM, Digital Threats: Research and Practice*.  
*Under Review*
- **Amith K. Belman** and Vir V. Phoha, “Behavioral Biometrics : The Failure of Normality Assumption”. In: *IEEE Transactions on Information Forensics and Security*.
- **Amith K. Belman**, Li Wang, S. S. Iyengar, Paweł Śniatała, Robert Wright, Robert Dora, Jacob Baldwin, Zhanpeng Jin, Vir V. Phoha, “Collecting and Sharing a Large Behavioral Biometric Dataset: Insights from BB-MAS”. Intended venue: *IEEE Transactions on Biometrics, Behavior and Identity Science*.