Learning of Identity from Behavioral Biometrics for Active Authentication

A Thesis

Submitted to the Faculty

of

Drexel University

by

Lex Fridman

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy in Electrical and Computer Engineering

December 2014

Acknowledgements

I'd like to thank Dr. Moshe Kam for his guidance and friendship in research and life. I have been very lucky to have the chance to work with and learn from him. I would also like to thank Dr. Steven Weber for teaching me a rigorous attention to detail and for a contagious enthusiasm for research. Finally, I'd like to thank Dr. Rachel Greenstadt for her guidance on several exciting projects we got to work on together.

Dedications

To mom and dad.

Table of Contents

List of Figures

Abstract

Learning of Identity from Behavioral Biometrics for
Active Authentication

Lex Fridman
Advisor: Moshe Kam, PhD
Co-advisor: Steven Weber, PhD

In this work, we look into the problem of active authentication on desktop computers and mobile devices. Active authentication is the process of continuously verifying a person's identity based on the cognitive, behavioral, and physical aspects of their interaction with the device. In this work, we consider several representative modalities including keystroke dynamics, mouse movement, application usage patterns, web browsing behavior, GPS location, and stylometry. We implement a binary classifier for each modality and organize the classifiers as a parallel binary decision fusion architecture. The decisions of each classifier are fed into a decision fusion center (DFC) which applies the Chair-Varshney fusion rule to generate a global decision. The DFC minimizes the probability of error using estimates of each local classifier's false rejection rate (FAR) and false acceptance rate (FRR). We test our approach on two large datasets of 67 desktop computer users and 200 mobile device users. We are able to characterize the performance of the system with respect to intruder detection time and to quantify the contribution of each modality to the overall performance.

1. Introduction

The challenge of identity verification for the purpose of access control in distributed communication systems is the tradeoff between maximizing the probability of intruder detection, and minimizing the cost for the legitimate user in time, distractions, and extra hardware and computer requirements. In recent years, behavioral biometric systems have been explored extensively in addressing this challenge [6].

Behavioral biometric systems rely on computer interface devices such as the keyboard and mouse that are already commonly available with most computers, and are thus low cost in terms of having no extra equipment requirements. However, their performance in terms of detecting intruders, and maintaining a low-distraction human-computer interaction (HCI) experience has been mixed [12], showing error rates ranging from 0% [49] to 30% [50] depending on context, variability in task selection, and various other dataset characteristics.

The bulk of biometric-based authentication work focused on verifying a user based on a static set of data. This type of one-time authentication is not sufficiently applicable to a live multi-user environment, where a person may leave the computer for an arbitrary period of time without logging off. This context necessitates continuous authentication when a computer is in a non-idle state. Validated access is important on two levels: (1) locally, to protect the offline data on the computer being used, and (2) globally, to protect the data traveling on a secured distributed network of which the computer is a part of. To represent a real-world scenario where such an authentication system may be used, we created a simulated office environment in order to collect behavioral biometrics associated with typical human-computer interaction (HCI) by an office worker over a typical work week.

In this thesis, we consider two large real-world datasets. For the first dataset, we use the data collected in an office environment, consider a representative selection of behavioral biometrics, and show that through a process of fusing the individual decisions of classifiers based on those metrics, we can achieve better performance than that of the best classifier from our classifier set. Due to their heterogeneous nature, it stands to reason that a properly designed set of good classifiers would outperform a single classifier which is "best" under specific circumstances. Moreover, given the low cost of installing these application-level classifiers, this approach may prove to be a cost-effective alternative to classifiers based on physiological biometrics [31]. We consider twelve classifiers, each

falling in one of three biometrics categories: keystroke dynamics, mouse movement, and stylometry.

For the second dataset, we consider the problem of active authentication on mobile devices, where the variety of available sensor data is much greater than on the desktop, but so is the variety of behavioral profiles, device form factors, and environments in which the device is used. We study four representative modalities of stylometry (text analysis), application usage patterns, web browsing behavior, and physical location of the device. In the remainder of the paper these four modalities will be referred to as TEXT, APP, WEB, and LOCATION, respectively. We consider the trade-off between intruder detection time and detection error as measured by false accept rate (FAR) and false reject rate (FRR). The analysis is performed on a dataset collected by the authors of 200 subjects using their personal Android mobile device for a period of at least 30 days. To the best of our knowledge, this dataset is the first of its kind studied in active authentication literature, due to its large size [19], the duration of tracked activity [45], and the absence of restrictions on usage patterns and on the form factor of the mobile device. The geographical colocation of the participants, in particular, makes the dataset a good representation of an environment such as a closed-world organization where the unauthorized user of a particular device will most likely come from inside the organization.

We propose to use decision fusion in order to integrate the classifier bank and make serial authentication decisions. While we consider here specific twelve classifiers, the strength of our decision-level approach is that additional classifiers can be added to the classifier bank without having to change the basic fusion rule, and with only minimal performance information required about the added classifiers. Moreover, it is easy to evaluate the marginal improvement of any added classifier to the overall performance of the system.

We evaluate the multimodal continuous authentication system on two large real-world datasets. We consider several parameters and metrics in presenting the system's performance. First, we look at the false acceptance rate (FAR) and the false rejection rate (FRR) when the decisions from each of the twelve classifiers are combined in the decision fusion center (DFC). Second, we assess the relative contribution of each individual classifier to the performance of the overall decision. Third, we observe the tradeoff between the time to first authentication decision and the error rates.

2. Related Work

2.1 Multimodal Biometric Systems

A defining problem of active authentication arises from the fact that a verification of identity must be carried out continuously on a sample of classifier data that varies drastically with time. The classification therefore has to be made based on a "window" of recent data, dismissing or heavily discounting the value of older data outside that window. Depending on what task the user is engaged in, some of the biometric classifiers may provide more data than others. For example, as the user browses the web, the mouse-related classifiers will be actively flooded with data, while the keystroke dynamics and stylometry classifiers may only get a few infrequent key press events. This motivates the recent work on multimodal authentication systems where the decisions of multiple classifiers are fused together [57]. In this way, the verification process is more robust to the dynamic mode of real-time HCI. The current approaches to the fusion of classifiers center around max, min, median, or majority vote combinations [38]. When neural networks are used as classifiers, an ensemble of classifiers is constructed and fused based on different initialization of the neural network [18].

Several active authentication studies have utilized multimodal biometric systems but have all, to the best of our knowledge: (1) considered a smaller pool of subjects, (2) have not characterized the temporal performance of intruder detection, and (3) have shown overall significantly worse performance than that achieved in our study. In particular, [23] have looked at similar classes of biometrics: keyboard dynamics, mouse movement, and stylometry. They used different features and classifiers, and did not propose a fusion scheme, but rather investigated each modality separately. The overall performance achieved ranged approximately from error rates of 0.1 to 0.4, which are significantly worse than the error rates achieved using the approach proposed in this thesis. Two fusion methods and a rich portfolio of features similar to the ones in this thesis were considered in [9] to achieve multi-modal authentication performance of 0.021 FAR and 0.024 FRR on a subject pool of 31 users. These error rates are an order of magnitude worse than those achieved in our work, and use a larger time window of 10 minutes.

Our approach in this thesis is to apply the Chair-Varshney optimal fusion rule [17] for the combination of available multimodal decisions. Furthermore, we are motivated by the work in [7] that greater reduction in error rates is achieved when the classifiers are distinctly different (i.e. using dif-

ferent behavioral biometrics). The strength of the decision-level fusion approach is that an arbitrary number of classifiers can be added without re-training the classifiers already in the system. This modular design allows for multiple groups to contribute drastically different classification schemes, each lowering the error rate of the global decision.

2.1.1 Mobile Active Authentication

With the rise of smartphone usage, active authentication on mobile devices has begun to be studied in the last few years. The large number of available classifiers makes for a rich feature space to explore. Ultimately, the question is the one that we ask in this thesis: what modality contributes the most to a decision fusion system toward the goal of fast, accurate verification of identity? Most of the studies focus on a single modality. For example, gait pattern was considered in [19] achieving an EER of 0.201 (20.1%) for 51 subjects during two short sessions, where each subject was tasked with walking down a hallway. Some studies have incorporated multiple modalities. For example, keystroke dynamics, stylometry, and behavioral profiling were considered in [55] achieving an EER of 0.033 (3.3%) from 30 simulated users. The data for these users was pieced together from different datasets. To the best of our knowledge, the dataset that we collected and analyzed is unique in all its key aspects: its size (200 subjects), its duration (30+ days), and the size of the portfolio of modalities that were all tracked concurrently with a synchronized timestamp.

2.2 Keystroke Dynamics and Mouse Movement

Keystroke dynamics is one of the most extensively studied topics in behavioral biometrics [37]. The feature space that has been investigated ranges from the simple metrics of key press interval [11] and dwell [26] times to multi-key features such as trigraph duration with an allowance for typing errors [12]. Furthermore, a large amount of classification methods have been studied for mapping these features into authentication decisions. Broadly, these approaches fall in one of two categories: statistical methods [63] and neural networks [14], with the latter generally showing higher FAR and FRR rates, but better able to train and make predictions on high-dimensional feature space.

While keyboard and mouse have been the dominant forms of HCI since the advent of the personal computer, mouse movement dynamics has not received nearly as much attention in the biometrics community in the last two decades as keystroke dynamics have. Most studies on mouse movement were either inconclusive due to small number of users [52] or required an excessively large static

corpus of mouse movement data to achieve good results [6], where an FAR and FRR of 0.0246 is achieved from a testing window of 2000 mouse actions. The work in [68] drastically reduces the size of the testing window to 20 mouse clicks. We base our selection of the three mouse metrics on their work but with more emphasis on mouse movement and not the mouse button presses.

One of the benefits of the mouse as behavioral biometric classifier is that it has a much simpler physical structure than a keyboard. Therefore, it is less dependent on the type of mouse and the environment in which the mouse is used. Keyboards, on the other hand, can vary drastically in size, response, and layout, potentially providing different biometric profiles for the same user. The simulated environment dataset we consider utilizes identical computer and working environment, so in our case, this particular robustness benefit is not important to authentication based on this data.

2.3 Stylometry

Authorship attribution based on linguistic style, or Stylometry, is a well-researched field [8, 54, 34, 42, 59, 32]. The main domain it is applied on is written language – identifying an anonymous author of a text by mining it for linguistic features. The theory behind stylometry is that everyone has a unique linguistic style ("stylome" [66]) that can be quantified and measured in order to distinguish between different authors. The feature space is potentially endless, with frequency measurements or numeric evaluations based on features across different levels of the text, including function words [47, 13], grammar [43], character n-grams [60] and more. Although stylometry has not been used for active user authentication, its application to this sort of task brings higher level inspection into the process, compared to other lower level biometrics like mouse movements or keyboard dynamics [68, 10], discussed in the following sections.

The most common practice of stylometry is in supervised learning, where a classifier is trained on texts of candidate authors, and used to attribute the stylistically closest candidate author to unknown writings. In an unsupervised setting, a set of writings whose authorship is unknown are classified into style-based clusters, each representing texts of some unique author.

In an active authentication setting, authorship verification is applied, where unknown text is classified by a unary author-specific classifier. The text is attributed to an author if and only if it is stylistically close enough to that author. Although pure verification is the ultimate goal, standard authorship attribution as a closed-world problem is an easier (and sometimes sufficient) goal. In either case, classifiers are trained in advance, and used for real-time classification of processed sliding

windows of input keystrokes. If enough windows are recognized as an author other than the real user, it should be considered as an intruder.

Another usage of stylometry is in author profiling [39, 8, 65, 27, 35] rather than recognition. Writings are mined for linguistic features in order to identify characteristics of their author, like age, gender, native language etc.

In a pure authorship attribution setting, where classification is done off-line, on complete texts (rather than sequences of input keystrokes) and in a supervised setting where all candidate authors are known, state-of-the-art stylometry techniques perform very well. For instance, at PAN-2012[1], some methods achieved more than 80% accuracy on a set of 241 documents, sometimes with added distractor authors.

In an active authentication setting, a few challenges arise. First, open-world stylometry is a much harder problem, with a tendency to high false-negative (false reject) rates. The unmasking technique [41] has been shown effective on a dataset of 21 books of 10 different 19^{th}-century authors, obtaining 95.7% accuracy. However, the amount of data collected by sliding windows of sufficiently small durations required for an efficient authentication system, along with the lack of quality coherent literary writings make this method perform insufficiently for our goal. Second, the inconsistent frequency nature of keyboard input along with the relatively large amount of data required for good performance of stylometric techniques make a large portion of the input windows unusable for learning writing style.

On the other hand, this type of setting allows some advantages in potential features and analysis method. Since the raw data consists of all keystrokes, some linguistic and technical idiosyncratic features can be extracted, like misspellings caught prior to being potentially auto-corrected and vanished from the dataset, or patterns of deletions (selecting a sentence and hitting delete versus repeatedly hitting backspace deleting character at-a-time). In addition, it is more intuitive in this kind of setting to consider overlap between consecutive windows, resulting with a large dataset, grounds for local voting based on a set of windows and control of the frequency in which decisions are outputted by the system.

Stylometry has been extensively applied to the problems of authorship attribution, identification, and verification. See [15] for a thorough summary of stylometric studies in each of these three problem domains along with their study parameters and the resulting accuracy. These studies traditionally use large sets of features (see Table II in [2]) in combination with support vector

[1] `http://pan.webis.de`

machines (SVMs) that have proven to be effective in high dimensional feature space [46], even in cases when the number of features exceeds the number of samples. Nevertheless, with these approaches, often more than 500 words are required in order to achieve adequately low error rates [25]. This makes them impractical for the application of real-time active authentication on mobile devices where text data comes in short bursts. While the other three modalities are not well investigated in the context of active authentication, this is not true for stylometry. Therefore, for this modality, we don't reinvent the wheel, and implement the n-gram analysis approach presented in [15] that has been shown to work sufficiently well on short blocks of texts.

2.3.1 Web Browsing, Application Usage, Location

Web browsing, application usage, and location have not been studied extensively in the context of active authentication. The following is a discussion of the few studies that we are aware of. Web browsing behavior has been studied for the purpose of understanding user behavior, habits, and interests [67]. Web browsing as a source for behavioral biometric data was considered in [5] to achieve average identification FAR/FRR of 0.24 (24%) on a dataset of 14 desktop computer users. Application usage was considered in [45], where cellphone data (from 2004) from the MIT Reality Mining project [21] was used to achieve 0.1 (10%) EER based on a portfolio of metrics including application usage, call patterns, and location. Application usage and movements patterns have been studied as part of behavioral profiling in cellular networks [61, 28, 45]. However, these approaches use position data of lower resolution in time and space than that provided by GPS on smartphones. To the best of our knowledge, GPS traces have not been utilized in literature for continuous authentication.

3. Authentication on Desktop Computers

3.1 Overview

Using the data collected in an office environment, we consider a representative selection of behavioral biometrics, and show that through a process of fusing the individual decisions of sensors based on those metrics, we can achieve better performance than that of the best sensor from our sensor set. Due to their heterogeneous nature, it stands to reason that a properly designed set of good sensors would outperform a single sensor which is "best" under specific circumstances. Moreover, given the low cost of installing these application-level sensors, this approach may prove to be a cost-effective alternative to sensors based on physiological biometrics [31]. We consider twelve sensors, each falling in one of three biometrics categories: keystroke dynamics, mouse movement, and stylometry. We evaluate the multimodal continuous authentication system on this large real-world dataset. We consider several parameters and metrics in presenting the system's performance. First, we look at the false acceptance rate (FAR) and the false rejection rate (FRR) when the decisions from each of the twelve sensors are combined in the decision fusion center (DFC). Second, we assess the relative contribution of each individual sensor to the performance of the overall decision. Third, we observe the tradeoff between the time to first authentication decision and the error rates. Fourth, we consider adversarial attacks on the system in the form of sensor "spoofing," and show that the system is robust to partial spoofing.

3.2 Dataset

The source of behavioral biometrics data we utilized for testing multi-modal fusion for the task of active authentication comes from a simulated work environment. In particular, we put together an office space, organized and supervised by a subset of the authors. We placed five desks in this space with a laptop, mouse, and headphones on each desk. This equipment and supplies were chosen to be representative of a standard office workplace. One of the important properties of this dataset is that of uniformity. Due to the fact that the computers and input devices in the simulated office environment were identical, the variation in behavioral biometrics data can be more confidently attributed to variation in characteristics of the users.

During each of the sixteen weeks of the data collection we hired 5 temporary employees for 40

hours of work. Each day they were assigned two tasks. The first was an open-ended blogging task, where they were instructed to write blog-style articles related in some way to the city in which the testing was carried out. This task was allocated 6 hours of the 8 hour workday. The second task was less open-ended. Each employee was given a list of topic or web articles to write a summary of. The articles were from a variety of reputable news sources, and were kept consistent between users except for a few broken links due to the expired lifetime of the linked pages. This second task was allocated 2 hours of the 8 hour workday.

Both tasks encouraged the workers to do extensive online research by using the web browser. They were allowed to copy and paste content, but they were instructed that the final work they produced was to be of their own authorship. As expected, the workers almost exclusively used two applications: Microsoft Word 2010 for word processing and Internet Explorer for browsing the web.

While the tasks were specified and suggested a combination of online research and word processing, the resulting behavior patterns were quite different. The productivity of workers, as measured by the number of words typed, varied drastically. They were purposefully not graded nor encouraged to be more productive, and therefore, tended to spend a large amount of their time browsing the web like they would outside of work: pursuing various interests, writing emails, commenting and chatting on Facebook and other social networks. In this way, the data we collected is representative of broader computer use than simply writing a blog on a particular subject. Each subject's interests and concerns outside of work had significant impact on their interaction with the computer.

Some of the users did not show up for work on one or more days. There were also several days on which the tracking software was shutdown prematurely for a user. Therefore, there were a few users for who the amount of data collected was significantly lower than the median. Therefore, we only used data from users who had over 54,000 seconds (15 hours) of *active* interaction with the computer. Before filtering out users in this way, we removed idle period in the data stream, where "idle" is defined as a period where neither the mouse nor keyboard were used for longer than 2 minutes. All such periods were shrunk down to 2 minutes. Therefore, due to such a temporal compression of the data, the 54,000 second threshold is based on active interaction with the computer. In this way we reduced the number of users in the dataset under consideration in this work from 80 down to 67.

Three data files produced by two tracking applications. They contain the following data:

- Mouse movement, mouse click, and mouse scroll wheel events at a granularity of 5 milliseconds.
- Keystroke dynamics (include press, hold, release durations) for all keyboard keys including

Metric	Total	Per User
Mouse move events	34,626,337	516,811
Mouse clicks	628,862	9,386
Scroll wheel events	404,531	4,397
Keystroke events	1,243,286	13,514

Table 3.1: Statistics on the 67-user subset of the biometric data contained in the dataset.

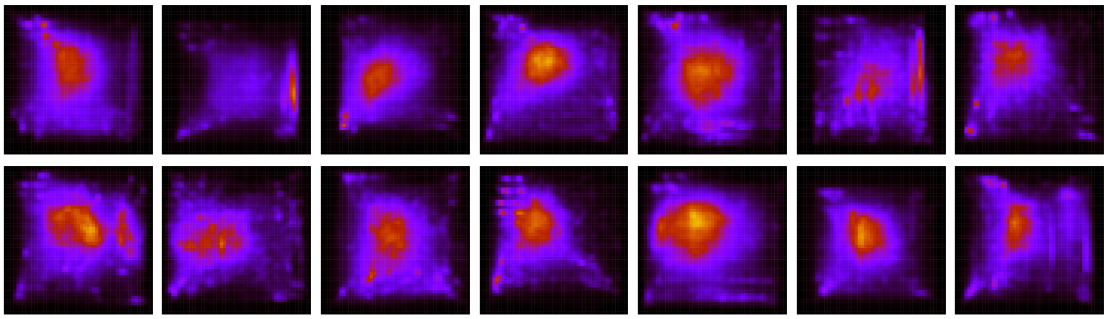

Figure 3.1: Each of the above subfigures is a visualization of aggregate mouse movement for one of the 67 users on their first day. We are only presenting 14 of the 67 users. This heat map is constructed by mapping the mouse movement data from the associated user to a 50 by 50 cell square image. The brighter the intensity of the cell, the more visits are recorded in that area of the screen. These figures visualize the intuition that there are distinct differences in the way each individual user interacts with the computer via the mouse.

special keys at a granularity of 5 milliseconds.

- Mapping of keys pressed to the application in focus at the time of the keyboard's use as input. The granularity for this data is 1 second but by synchronizing with the data from the first two streams, higher resolution timing information can be inferred.

Table 3.1 shows statistics on the biometric data in the corpus. The table contains data aggregated over all 67 users. It also shows the average amount of data available per user. The keystroke events include both the alpha-numeric keys and also the special keys such as `shift`, `backspace`, `ctrl`, `alt`, etc. In counting the key presses and the mouse clicks for Table 3.1, we count just the down press and not the release.

As an example of the variation in the dataset, Fig. 3.1 shows a heat map visualization of the aggregate first-day mouse movements for 14 of the 67 users. It provides an intuition that the users have unique behavioral profiles of interaction with the computer via the mouse to a degree that distinct patterns emerge even in heat maps that aggregate a full day's worth of data. Some users

spend a lot of time on the scroll bar, some users focus their attention to the top left of the screen, and some users frequently move their mouse big distances across the screen.

3.3 Behavioral Biometric Modalities

The sets of features we consider in this thesis are linguistic style (stylometry), mouse movement patterns, and keystroke dynamics. We construct classifiers (or classifiers) from these features differently depending on the feature. For keystroke dynamics and mouse movement features, each individual feature is tracked by one classifier that uses a Naive Bayes classifier [58]. For stylometry, the portfolio of features is combined into one classifier using support vector machines (SVMs) [16]. Each of these types of classifiers work differently in terms of required amount of input data, type of collected data (mouse events, keystroke event) and performance.

We broadly categorize the classifiers in this thesis according to the degree of conscious cognitive involvement measured by the classifiers. The distinction can be thought of as that between "how" and "what". We refer to the mouse movement and keystroke dynamics classifiers as "low-level", since they measure *how* we use the mouse and *how* we type. On the other hand, the website domain frequency and stylometry classifiers are "high-level" because they track *what* we click on with the mouse and *what* we type. Table 3.2 shows the twelve classifiers under consideration in this thesis. The frequency listed is an upperbound on frequency that a classifier produces a classification. The actual frequency depends on the time-based windows size that the classifiers is configured to use in training and testing phases.

3.3.1 Keystroke Dynamics and Mouse Movement

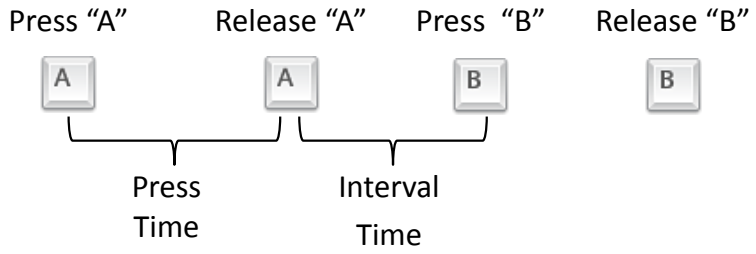

Figure 3.2: The keystroke dynamics metrics are computed from the time between the press and release event and vice versa.

Metric	Frequency (Hz)
1. Key Press Duration	0.1295
2. Key Interval	0.1248
3. Mouse Curve Distance	1.0271
4. Mouse Curve Curvature	0.7153
5. Mouse Button Press Duration	0.0423
6. Mouse Click-Path Speed	0.0385
7. Mouse Click-Path Wandering	0.0385
8. Mouse Click-Path Angle	0.0385
9. Mouse Nonclick-Path Speed	0.0201
10. Mouse Nonclick-Path Wandering	0.0201
11. Mouse Nonclick-Path Angle	0.0201
12. Stylometry	0.1295

Table 3.2: The classifiers whose performance is investigated in this thesis. These include 1 stylometry, 2 keystroke, and 9 mouse classifiers. For each classifier, listed is the average frequency across all 67 users that an event associated with that classifier is observed during active interaction with the computer.

For any change in the position of the mouse, the raw data received from the mouse tracker are (1) the pixel coordinates of the new position and (2) the delay in milliseconds between the recording of this new position and the previously recorded action. Usually that delay is 5 milliseconds, but sometimes the sampling frequency degrades for short periods of time. This tuplet gives us the basic data element based on which all the mouse movement metrics are computed (given an initial position on the screen).

In this thesis, we consider nine mouse-based metrics as listed in Table 3.2, and illustrated in Fig. 3.3. A "mouse curve" is an uninterrupted sequence of three mouse move events. A "mouse path" is an uninterrupted sequence of mouse move events with other type of events before and after it. A "click path" is a mouse path that ends in a mouse button click. Conversely, a "nonclick path" is a mouse path that ends in an event other than a mouse button click. The mouse classifiers are based on features of these sequences of mouse events.

We chose two of the simplest and most frequently occurring keystroke dynamics features as illustrated in Fig. 3.2: (K1) the interval between the release of one key and the press of another and (K2) the dwell time between the press of a key and its release. While the dwell time K2 is a strictly positive number, the interval K1 can be negative if another key is pressed before a prior one is released.

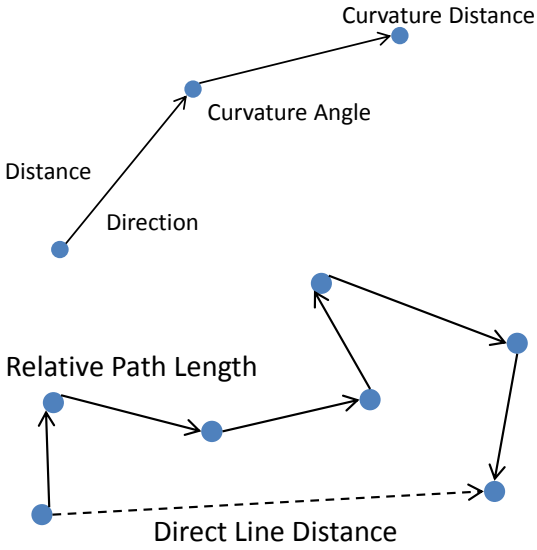

Figure 3.3: The mouse movement metrics are computed from a set of continuous move events (defined by positions on the virtual screen). On the left are three points that define a "mouse curve" and based on which the mouse curve distance and curvature metrics are computed. On the right are 3 or more points that define a "mouse path" and based on which the mouse path speed, angle, and "wandering" metrics are based.

3.3.2 Stylometry

We chose the setting of closed-world stylometry: we developed classifiers trained on the closed set of users. The classifier's output is the author to which the text is attributed.

In the preprocessing phase, we parsed the keystrokes log files to produce a list of documents consisting of overlapping windows for each user, with the following time-based sizes (in seconds): 10, 30, 60, 300, 600 and 1,200. For the first 3 settings we advanced the sliding window with steps of 10 seconds, and for the last 3 – steps of 60 seconds. The step size determines how often a decision can be made by the classifier.

During preprocessing, only keystrokes were considered and all special keys were converted to unique single-character placeholders. For instance `BACKSPACE` was converted to β and `PRINTSCREEN` was converted to π. Any representable special keys like `\t` and `\n` were taken as is (i.e. tab and newline, respectively).

The constructed feature set, denoted the *AA* feature set hereinafter, is a variation of the *Writeprints* [3] feature set, which includes a vast range of linguistic features across different levels of text. A summarized description of the features is presented in Table 3.3. By using a rich linguistic feature set we hope to capture the user's writing style. With the special-character placeholders, some features

capture aspects of the user's style usually not found in standard authorship problem settings. For instance, frequencies of backspaces and deletes provide some evaluation of the user's typo-rate.

The features were extracted using the *JStylo* framework [1] [46], an open-source authorship attribution platform. JStylo was chosen since it is equipped with fine feature definition capabilities. Each feature is uniquely defined by a set of its own document preprocessing tools, one unique feature extractor (the core of the feature), feature postprocessing tools, and normalization/factoring options. The features available in JStylo are either frequencies of a class of related features (e.g., frequencies of "a", "b", ..., "z" for the "letters" feature class) or some numeric evaluation of the input document (e.g., average word length, or Yule's Characteristic K). Its output is compatible with the data mining and machine learning platform Weka [29], which we used for the classification process.

Group	Features
Lexical	Avg. word-length
	Characters
	Most common character bigrams
	Most common character trigrams
	Percentage of letters
	Percentage of uppercase letters
	Percentage of digits
	Digits
	2-digit numbers
	3-digit numbers
	Word length distribution
Syntactic	Function words
	Part-of-speech (POS) tags
	Most common POS bigrams
	Most common POS trigrams
Content	Words
	Word bigrams
	Word trigrams

Table 3.3: The *AA* feature set. Inspired by the *Writeprints* [3] feature set, includes features across different levels of the text. Some features are normalized frequencies of feature classes; others are numerical evaluations of the input text.

Two important processing procedures were applied in the feature extraction phase. First, every word-based feature (e.g., the function words class, or different word-grams) was applied a tailor-made

[1] http://psal.cs.drexel.edu/

preprocessing tool developed for this unique dataset, that applies the relevant special characters on the text. For instance, the character sequence `ch`$\beta\beta$`Cch`$\beta\beta$`hicago` becomes `Chicago`, where β represents backspace. Second, since the windows are determined by time and not amount of collected data, normalization is crucial for all frequency-based features (which consist the majority of the features).

For classification, we used sequential minimal optimization (SMO) support vector machines [51] with polynomial kernel, available in Weka. Support vector machines are commonly used for authorship attribution [1, 40, 69] and known to achieve high performance and accuracy.

Finally, the data was analyzed with the stylometry classifiers using a varying threshold for minimum characters-per-window to consider, spanning from 100 to 1000 with steps of 100. For every threshold set, all windows with less than that amount of characters were thrown away, and for those windows the classifier output was "no decision". The different thresholds allow us to assess the tradeoff in the classifier's performance in terms of accuracy and availability: as the threshold increases, the window is richer with data and will potentially be classified with higher accuracy, but the portion of total windows that pass the threshold decreases, making the classifier less available. Fig. 3.4 illustrates the average percentage of usable windows, after removing all those that do not pass the minimum characters-per-window threshold.

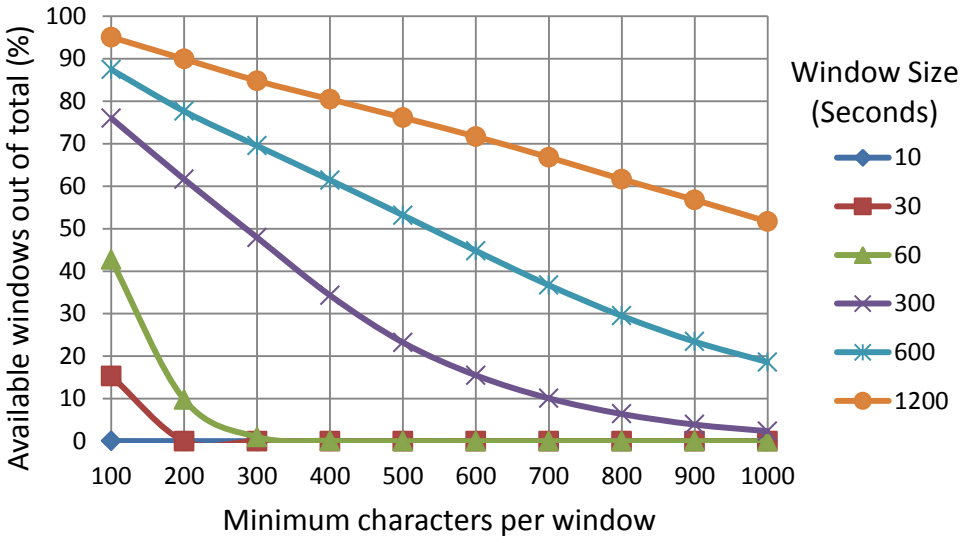

Figure 3.4: Percentage of remaining windows out of the total windows after filtering by the minimum characters-per-window threshold.

3.4 Decision Fusion

The motivation for the use of multiple classifiers to detect an event is to harness the power of the classifiers to provide an accurate assessment of a studied phenomenon, which a single classifier may not be able to provide. In centralized architectures, raw data from all classifiers monitoring the same space are communicated to a central point for integration, the fusion center. However quite often the use of a centralized architecture is not desirable or practical. The factor weighing against centralization is the need to transfer large volumes of data between local detector and fusion center. Another is the fact that in many systems specialized local detectors already exist, and it is more convenient to fuse their decisions rather than re-create the detection algorithms at the fusion center. In the distributed architectures, some processing of data is performed at each classifier, and the resulting information is sent out from each classifier to a central processor for subsequent processing and final decision making. On most scenarios significant reduction in required bandwidth for data transfer and modularity are the main advantages of this approach. The price is sub-optimality of the decision /detection scheme.

Decision fusion with distributed classifiers is described by Tenney and Sandell in [62] who studied a parallel decision architecture. As described in [36], the system comprises of n local detectors, each making a decision about a binary hypothesis (H_0, H_1), and a decision fusion center (DFC) that uses these local decisions $\{u_1, u_2, ..., u_n\}$ for a global decision about the hypothesis. The i^{th} detector collects K observations before it makes its decision, u_i. The decision is $u_i = 1$ if the detector decides in favor of H_1 (decision D_1), and $u_i = -1$ if it decides in favor of H_0(decision D_0). The DFC collects the n decisions of the local detectors through ideal communication channels and uses them in order to decide in favor of $H_0(u = -1)$ or in favor of $H_1(u = 1)$. Fig. 3.5 shows the architecture and the associated symbols. Tenney and Sandell [62] and Reibman and Nolte [53] studied the design of the local detectors and the DFC with respect to a Bayesian cost, assuming the observations are independent conditioned on the hypothesis. The ensuing formulation derived the local and DFC decision rules to be used by the system components for optimizing the system-wide cost. The resulting design requires the use of likelihood ratio tests by the decision makers (local detectors and DFC) in the system. However the thresholds used by these tests require the solution of a set of nonlinear coupled differential equations. In other words, the design of the local decision makers and the DFC are co-dependent. In most scenarios the resulting complexity renders the quest for an optimal design impractical.

Chair and Varshney in [17] developed the optimal fusion rule when the local detectors are fixed and local observations are statistically independent conditioned on the hypothesis. Data Fusion Center is optimal with respect to a Bayesian cost, given the performance characteristics of the local fixed decision makers. The result is a suboptimal (since local detectors are fixed) but computationally efficient and scalable design. In this study we use the Chair-Varshney formulation. As described in [36], the *Bayesian risk* $\beta^{(k)}(C_{00}, C_{01}, C_{10}, C_{11})$ is defined for the k^{th} decision maker in the system as

$$\begin{aligned}\beta^{(k)}(C_{00}, C_{01}, C_{10}, C_{11}) = \quad & C_{00}^{(k)} Pr(H_0, D_0) + C_{10}^{(k)} Pr(H_0, D_1) \\ & + C_{01}^{(k)} Pr(H_1, D_0) + C_{11}^{(k)} Pr(H_1, D_1)\end{aligned} \tag{3.1}$$

where $C_{00}^{(k)}, C_{01}^{(k)}, C_{10}^{(k)}, C_{11}^{(k)}$ are the prespecified cost coefficients of the k^{th} decision maker for each combination of hypothesis and detector decision: $C_{ij}^{(k)}$ is the cost incurred when the k^{th} decision maker decides D_i when H_j is true. For the cost combination $C_{00}^{(k)} = C_{11}^{(k)} = 0$ and $C_{01}^{(k)} = C_{10}^{(k)} = 1$, the Bayesian cost becomes the *probability of error*. We consider a suboptimal system where each detector $k = 1, 2, ..., n$ minimizes locally a Bayesian risk $\beta^{(k)}$ and the DFC ($k = 0$) is optimal with respect to $\beta^{(0)}$, given the local detector design. In the subsequent work, we assume $\beta^{(k)} = \beta^{(0)}$, $k = 1, 2, ..., n$ (all local detectors minimize the same Bayesian risk) and the superscript k is therefore omitted. Specifically we use throughout the thesis

$$\begin{aligned} C_{00}^{(k)} &= C_{11}^{(k)} = 0, k = 1, 2, ..., n \\ C_{10}^{(k)} &= C_{01}^{(k)} = 1, k = 1, 2, ..., n \end{aligned} \tag{3.2}$$

namely the local detectors and the DFC each minimizes the probability of error.

3.4.1 Fusion Rule

The parallel distributed fusion scheme (see Fig. 3.5) allows each classifier to observe an event, minimize the local risk and make a local decision over the set of hypothesis, based on only its own observations. Each classifier sends out a decision of the form:

$$u_i = \begin{cases} 1, & \text{if } H_1 \text{ is decided} \\ -1, & \text{if } H_0 \text{ is decided} \end{cases} \tag{3.3}$$

The fusion center combines these local decisions by minimizing the global Bayes' risk. The

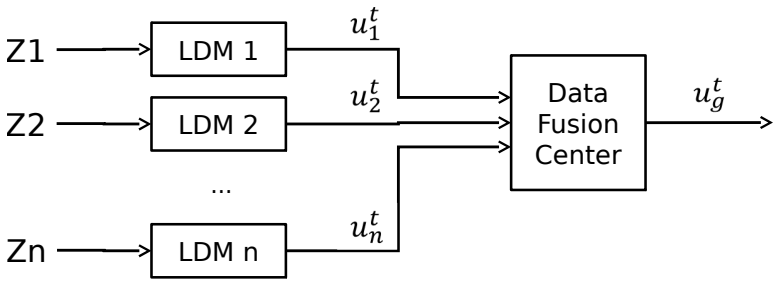

Figure 3.5: Architecture for the fusion of decentralized detectors.

optimum decision rule performs the following likelihood ratio test

$$\frac{P(u_1,...,u_n|H_1)}{P(u_1,...,u_n|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0(C_{10}-C_{00})}{P_1(C_{01}-C_{11})} = \tau \tag{3.4}$$

where the a priori probabilities of the binary hypotheses H_1 and H_0 are P_1 and P_0 respectively and C_{ij} are the costs as defined previously. For costs as defined in (3.2), the Bayes' risk becomes total probability of error and the right hand side of (3.4) becomes $\frac{P_0}{P_1}$. In this case the general fusion rule proposed in [17] is

$$f(u_1,...,u_n) = \begin{cases} 1, & \text{if } a_0 + \sum_{i=0}^{n} a_i u_i > 0 \\ -1, & \text{otherwise} \end{cases} \tag{3.5}$$

with P_i^M, P_i^F representing the *False Rejection Rate* (FRR) and *False Acceptance Rate* (FAR) of the i^{th} classifier respectively. The optimum weights minimizing the global probability of error are given by

$$a_0 = \log \frac{P_1}{P_0} \tag{3.6}$$

$$a_i = \begin{cases} \log \frac{1-P_i^M}{P_i^F}, & \text{if } u_i = 1 \\ \log \frac{1-P_i^F}{P_i^M}, & \text{if } u_i = -1 \end{cases} \tag{3.7}$$

Kam et al. in [36] developed expressions for the the global performance (global FAR and FRR) of the distributed system described above. Exact expressions for global error rates are given in [36].

The threshold in (3.4) requires knowledge of the a priori probabilities of the hypotheses. In practice, these probabilities are not available, and the threshold τ is determined using different considerations (such as fixing the probability of false alarm of the DFC).

3.4.2 Extendable Fusion Framework

As is explain in §3.4.1, the performance of the fused global detector improves as the number of local classifiers increases. Furthermore, it is shown in [7] that fusion of classifiers trained on distinct feature sets leads to greatest reduction in system error. In our context, the ideal active authentication system gathers input from as many different behavioral biometric classifiers as possible. In designing the fusion system one of our goals was to provide a straightforward way of adding classifiers to the system without having to change algorithms and with simple and uniform characterization of each classifier. In fact our formulation requires only that the FAR and FRR be supplied, so that they can be incorporated in (4.5) and (4.6).

3.5 Results

3.5.1 Training, Characterization, Testing

The data of each of the 67 users' active interaction with the computer was divided into 5 equal-size folds (each containing 20% time span of the full set). We performed training of each classifier on the first three folds (60%). We then tested their performance on the fourth fold. This phase is referred to as "characterization", because its sole purpose is to form estimates of FAR and FRR for use by the fusion algorithm. We then tested the performance of the classifiers, individually and as part of the fusion system, on the fifth fold. This phase is referred to as "testing" since this is the part that is used for evaluation the performance of the individual classifiers and the fusion system. The three phases of training, characterization, and testing as they relate to the data folds are shown in Fig. 4.4.

- Training on folds 1, 2, 3. Characterization on fold 4. Testing on fold 5.
- Training on folds 2, 3, 4. Characterization on fold 5. Testing on fold 1.
- Training on folds 3, 4, 5. Characterization on fold 1. Testing on fold 2.
- Training on folds 4, 5, 1. Characterization on fold 2. Testing on fold 3.

- Training on folds 5, 1, 2. Characterization on fold 3. Testing on fold 4.

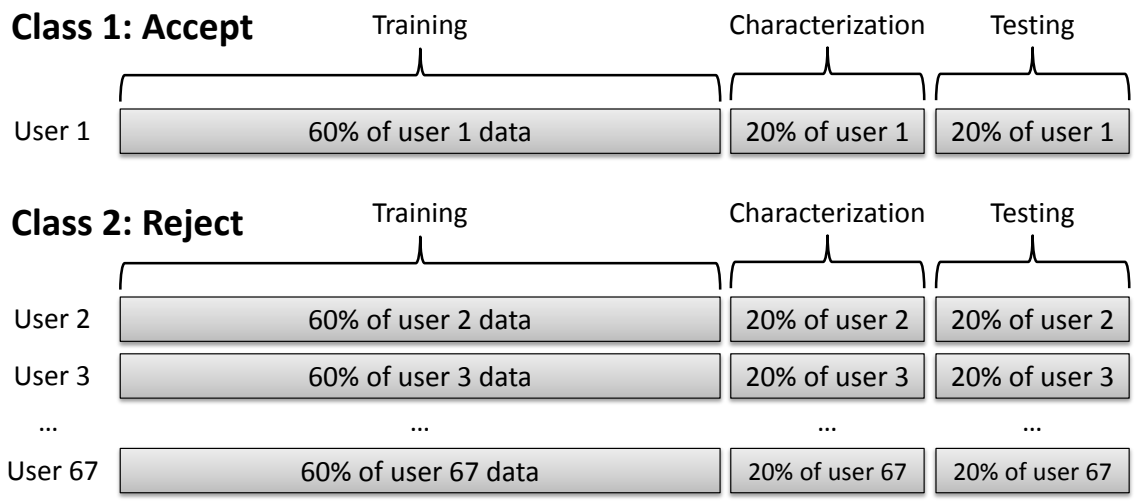

Figure 3.6: The three phases of processing the data to determine the individual performance of each classifiers and the performance of the fusion system that combines some subset of these classifiers.

The common evaluation method used with each classifier for data fusion was measuring the averaged error rates across five experiments; In each experiment, data of 3 folds was taken for training, 1 fold for characterization, and 1 for testing. The FAR and FRR computed during characterization were taken as input for the fusion system as a measurement of the expected performance of the classifiers. Therefore each experiment consisted of three phases: 1) train the classifier(s) using the training set, 2) determine FAR and FRR based on the training set, and 3) classify the windows in the test set.

Unless otherwise specified, the experiments we ran were using the fusion system on the full 67 user set with the 2 keystroke dynamics classifiers, 9 mouse classifiers, and the stylometry classifier.

3.5.2 Contribution of Individual Classifiers

For each low-level classifier, we used the Naive Bayes classifier [33] for mapping from the feature space to the decision space. For the stylometry classifier, we used an SVM as described in §3.3. In the training phase for low-level classifiers, the empirical distribution for feature probabilities were constructed from the frequency of each feature in the training segment of each user's data. Two such histograms were constructed for each user j. The first histogram was constructed from the

training segment of the data of that user. The second histogram was constructed from all the training segments of the other users. These two histograms are the empirical feature distributions associated with each user.

In the characterization and testing phases, for each user and each metric, the Naive Bayes Classifier considered a collection of events $\Omega = \{x_t | T_{\text{current}} - T(x_t) \leq \omega\}$ where ω is a fixed window size in seconds, $T(x_t)$ is the timestamp of event x_t, and T_{current} is the current timestamp. The maximum a posteriori (MAP) rule was then used to pick the most likely hypothesis:

$$H^* = \underset{i \in \{0,1\}}{\operatorname{argmax}} P(H_i) \prod_{x_t \in \Omega} P(x_t | H_i), \tag{3.8}$$

where H_1 is the "authentic" class, H_0 is the "non-authentic" class, as discussed in §4.3.2, and H^* is the most likely class associated with the observed biometric data. Unless otherwise stated we assume $P(H_0) = P(H_1) = 0.5$. The feature probability $P(x_t|H_i)$ is estimated by a non-parametric distribution formed in the training phase.

Fig. 4.5 shows the FAR and FRR rates respectively for the 11 keystroke and mouse movement classifiers. For all four figures, the performance is averaged over 67 users and characterized with respect to the time-window size used by each of the classifiers. Any data older than the duration of the window is discarded. The classifier only provides a decision when the time-window includes a minimum amount of events. For both mouse and keyboard that threshold was set to 5 events. As the size of the decision window increases, the FAR and FRR rates generally decrease for all classifiers. The performance of the individual classifiers varies from error rates as low as 0.01 to above 0.3.

The absolute performance of the fusion system is presented §3.5.3, but first we look at the contribution of each of the 11 low level classifiers of keystroke dynamics and mouse movement to the overall performance of the fusion system. We measure this relative contribution C_i by evaluating the performance of the system with and without the classifier, and computing the contribution by:

$$C_i = \frac{E_i - E}{E_i} \tag{3.9}$$

where E is the error rate computed by averaging FAR and FRR of the fusion system using the full portfolio of 11 low-level classifiers, E_i is the error rate of the fusion system using all but the i-th classifier, and C_i is the relative contribution of the i-th classifier as shown in Fig. 4.8.

Classifiers based on the features of mouse curve distance, mouse curve curvature, key press duration, and key interval contributed the most to the fused decision. This can be explained by

the fact that these four metrics are also those that appear with the highest frequency. Therefore, while their error rates individually are not always the lowest, the frequency of their "firing" makes up for a higher error rate when backed by the portfolio of the other classifiers. On a time scale of 60 to 120 seconds where the low-level classifiers excel, the stylometry classifier performed poorly and contributed almost zero to the overall decision, and thus was not included in the figure. The stylometry classifier begins contributing considerably on a longer time scale of 10 to 30 minutes.

3.5.3 Time to First Decision

Two conflicting metrics of an active authentication system are response-time and performance. The less the system waits before making an authentication decision, the higher the expected rate of error. As more keystroke and mouse events trickle in, the system can refine its classification decision from an initial "neutral" stance of FAR = FRR = 0.5. In Fig. 3.11, we show the tradeoff between the decision time and performance.

The "time to first decision" is the time between the first keyboard or mouse event and the first decision produced by the fusion system. This metric can be thought of as "decision window size". Events older than the time range covered by the time-window are disregarded in the fused decision. As describe in §3.5.2 when a decision window contains less than 5 events, no decision is produced by the fusion system.

As the size of the decision window increases, the performance of the system improves, dropping below 0.01 FAR and FRR in 30 seconds as shown in Fig. 3.11. These plots also compare the performance of the fusion system on a 10 user subset and the full 67 user dataset. Performance degrades but not significantly and gives promise to the scalability of the system in the closed world environment.

When the user of the system changes, a decision window will contain a mix of events from two different users. In Fig. 3.12 the second user is an "intruder". The decision value "+1" corresponds to a valid user. The decision value "-1" corresponds to an intruder. The figure shows the real-time detection of an intruder based on two different decision windows of 10 seconds and 100 seconds. The complete detection period in this case is approximately equal to twice the decision window because both the individual classifiers and the fusion system are using the same size window. For example, for a 100 second window, it is not until 100 seconds after the intruder enters that classifiers are operating purely on the data received from the intruder and not on the previous user. It's not until 200 seconds after the intruder enters that the fusion system integrates classifier data based purely

on the intruder interaction with the computer.

3.5.4 Robustness to Partial Spoofing

"Partial spoofing" is the successful mimicking of a valid user by an adversary on a subset of classifiers contributing to the fused decision. The result is that the spoofed classifiers incorrectly classify the current user as the valid user. We emulate this form of perfect spoofing by feeding valid user data to the classifiers marked as "spoofed". Fig. 3.13 shows how the performance of the system degrades with an increasing number of spoofed classifiers, in order from highest-contributing to lowest as shown in Fig. 4.8. In other words, *mouse curve distance* was spoofed first, *mouse curve curvature* was spoofed second, and so on. The performance of the partially-spoofed fusion system is evaluated using the FAR metric, since what is being measured is the rate at which the system incorrectly identifies an intruder as a valid user. The same classifiers and fusion system described in §3.5.3 were used to generate the results in this section.

3.5.5 Closed World Versus Open World

The behavioral biometrics dataset considered in this thesis is constrained in that all the users were performing a similar task for a similar period of time on exactly the same desk, keyboard, mouse, and computer. This removed variability in the office environment as a factor in the biometric footprint of each user. Furthermore, we used the critical assumption of a "closed world": no one other than the 67 users in the dataset will never seek to use the computers under the protection of our authentication system. In other words, every user in the system contributed a significant amount of biometric data to the training process.

Naturally, the question arises how well the system performs when a 68'th user is injected in the system, without participating in the training. While we can't answer that exact question, we can do so for a subset of the data by removing some of the users from the training but still using them in the testing group. More precisely, we run the following experiment:

- Train on m users.
- Test on the same m users. The results of this testing phase are labeled "Closed: m users".
- Test on $2m$ users, m of which were part of the training set. The results of this testing phase are labeled "Open: m users".

The above process is repeated 10 times for random selections of $2m$ users to generate two curves in Fig. 3.14. The figure contains performance results for $m = 10$ and $m = 25$. The error rates increase significantly with the introduction of users who were not part of the training process. So while Fig. 3.11 indicates promise that the system is scalable under the closed world constraint, Fig. 3.14 indicates that the system is likely no longer scalable when this constraint is removed and user from outside the training environment are allowed to interact with the computers.

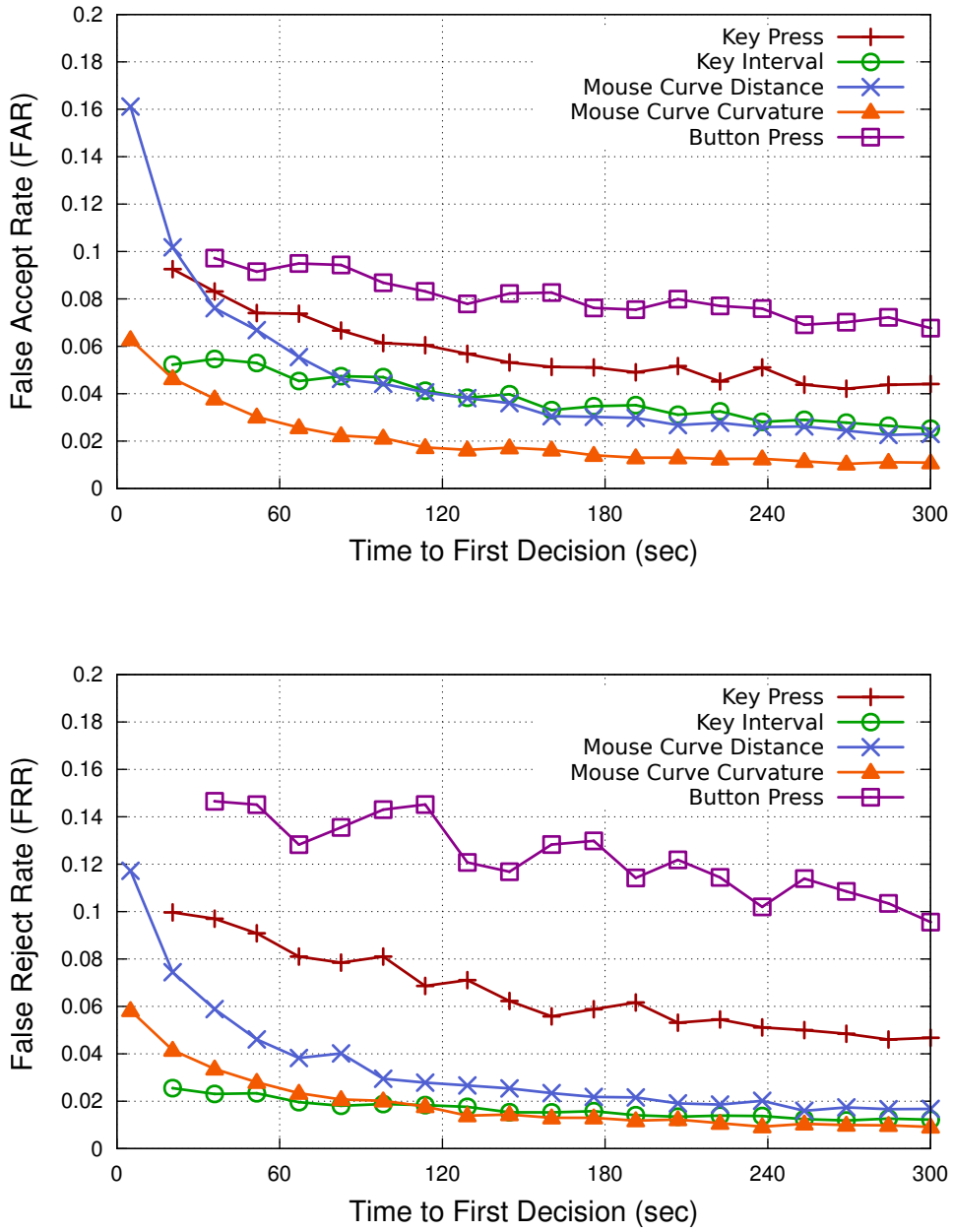

Figure 3.7: FAR and FRR performance of 5 of 11 keystroke dynamics and mouse movement classifiers. Note that the range of the plots for this set of classifiers is shorter (300 seconds) than for the set of classifiers in Fig. 3.8.

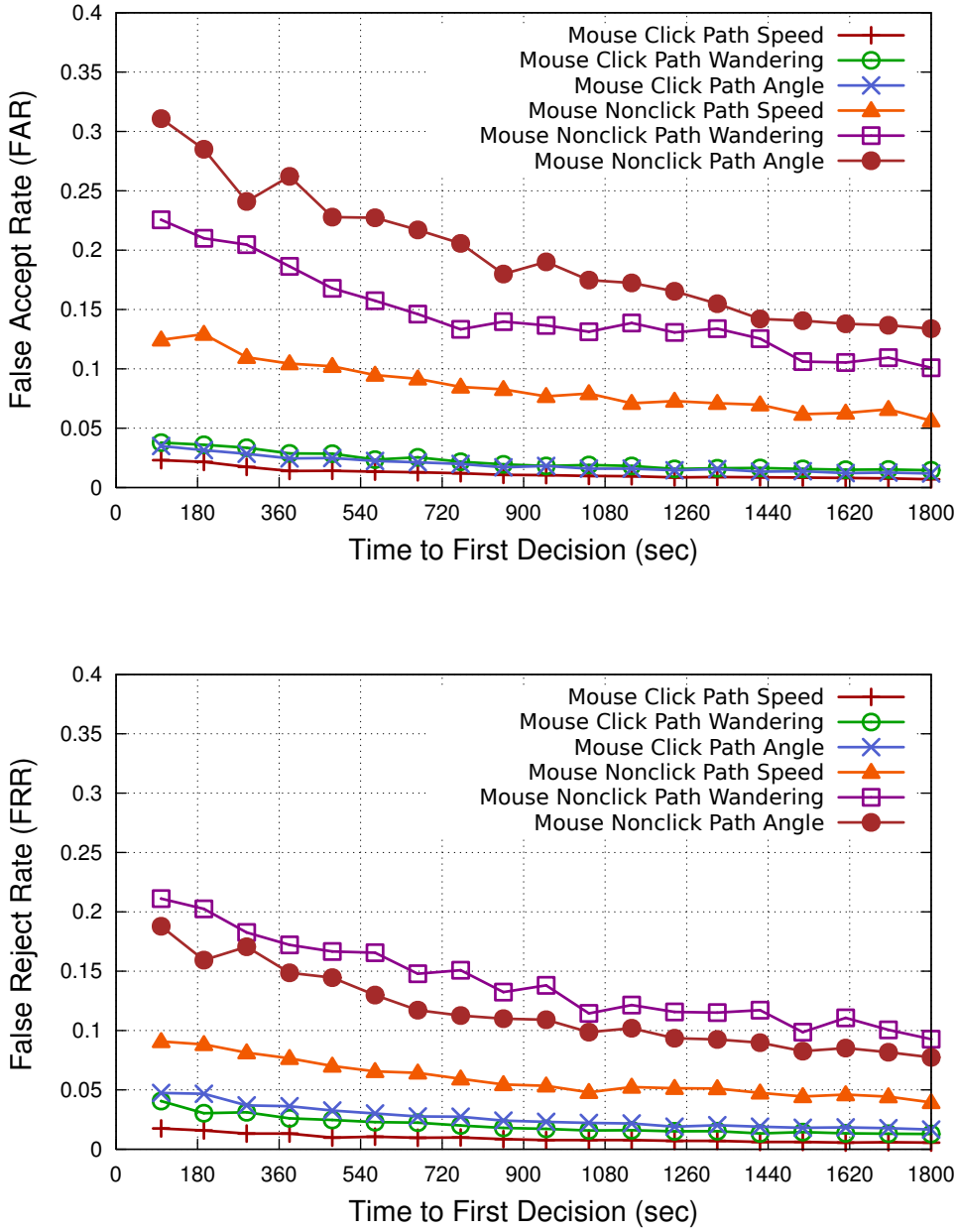

Figure 3.8: FAR and FRR performance of 5 of 11 keystroke dynamics and mouse movement classifiers. Note that the range of the plots for this set of classifiers is longer (1800 seconds) than for the set of classifiers in Fig. 4.5.

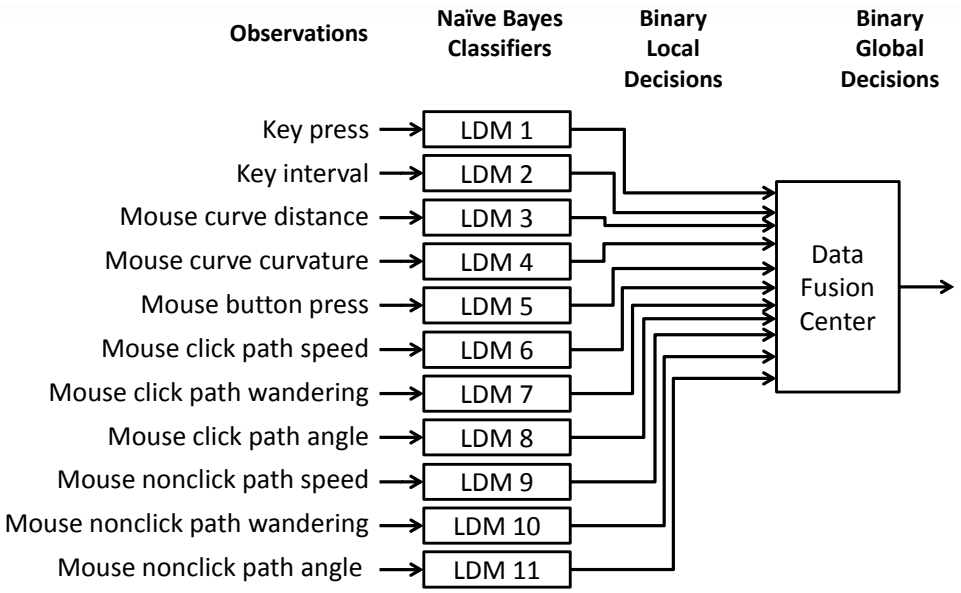

Figure 3.9: The portfolio of 11 low-level classifiers based on keystroke dynamics and mouse movement that forms the basis of our evaluations in §4.4.

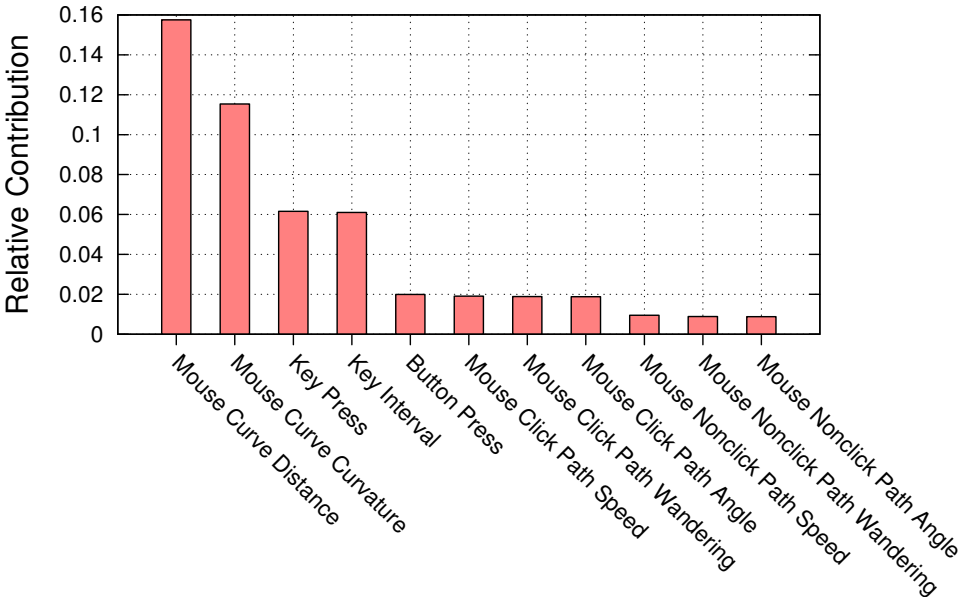

Figure 3.10: Relative contribution of each of the 11 low-level classifiers of keystroke dynamics and mouse movement to the fused decision. The contribution is computed according to (4.7).

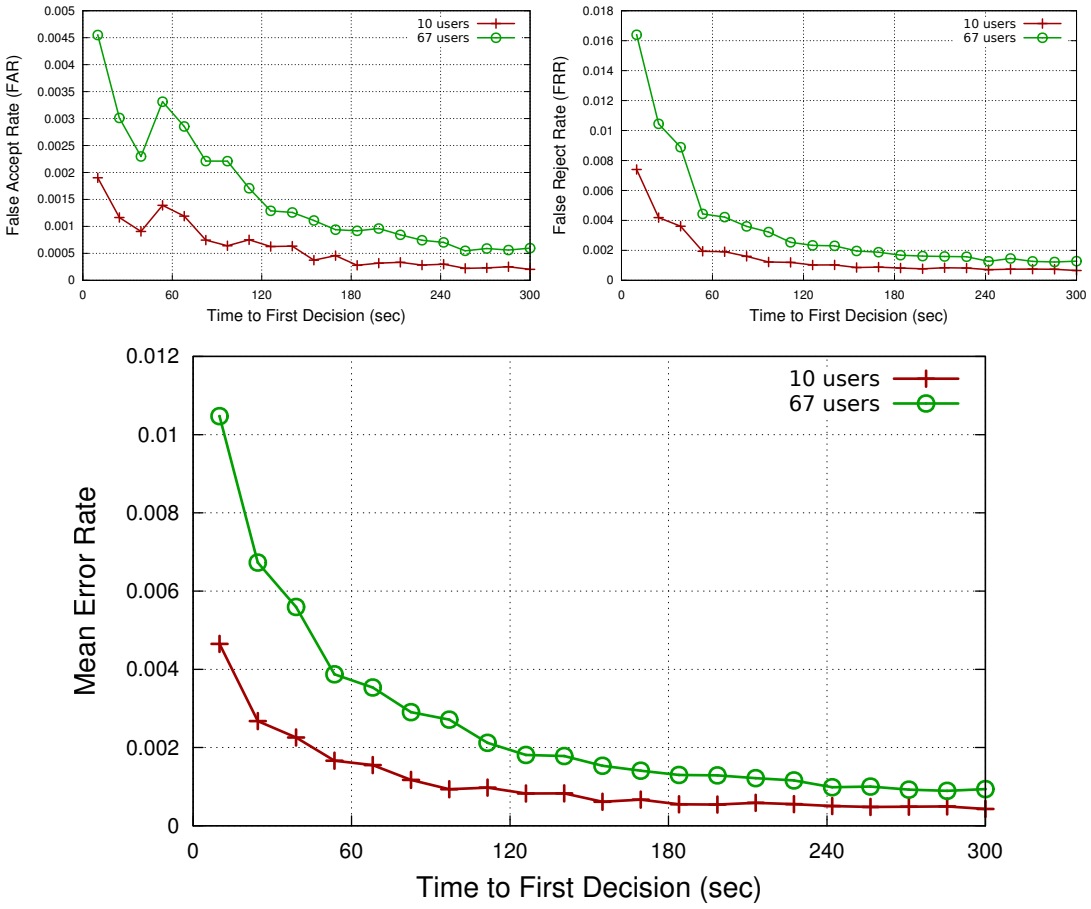

Figure 3.11: The performance of the fusion system of 11 low-level classifiers on the 10 users and 67 users. The standard deviation of each data point is small, with the coefficient of variation less than 0.5 for each point.

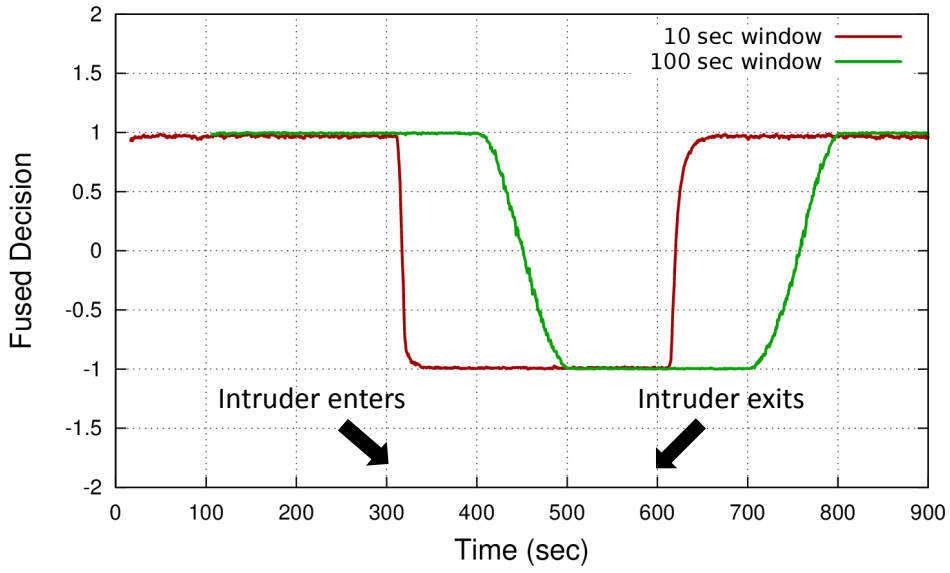

Figure 3.12: Visualization of the real-time detection of an intruder averaged over 10,000 random samples of data from the 67 user dataset. A decision value of 1 indicates that the system believe the user to be authentic, and -1 otherwise. Due to the low error rates of the fusion system, an intruder is successfully detected even with small time-window of 10 seconds.

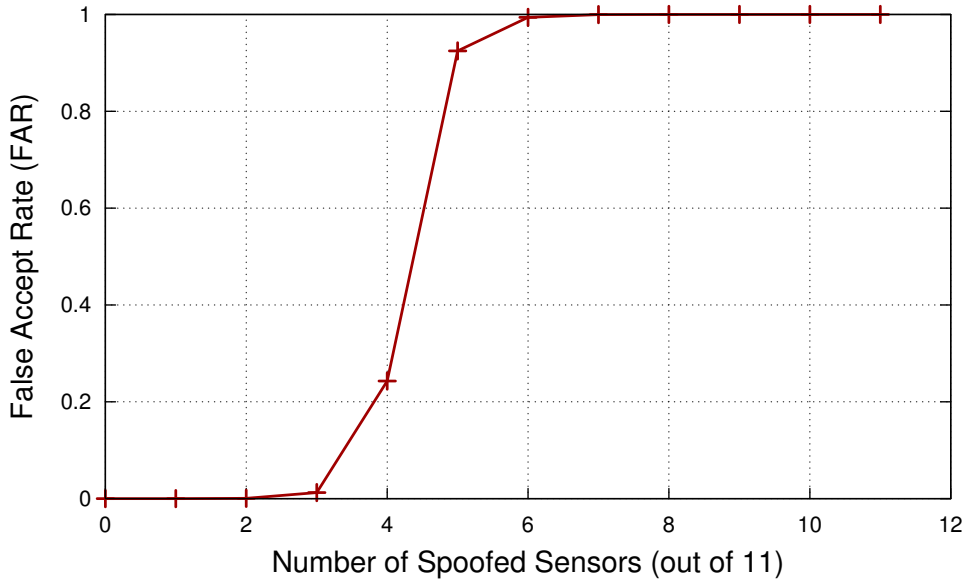

Figure 3.13: FAR of a partially-spoofed fusion system. The classifiers are compromised in the order of decreasing contribution as shown in Fig. 4.8. As the number of spoofed classifiers increases from 0 to 11, the performance of the system degrades from nearly 0 to nearly 1 FAR. The standard deviation of each data point is small, with the coefficient of variation less than 0.5 for each point.

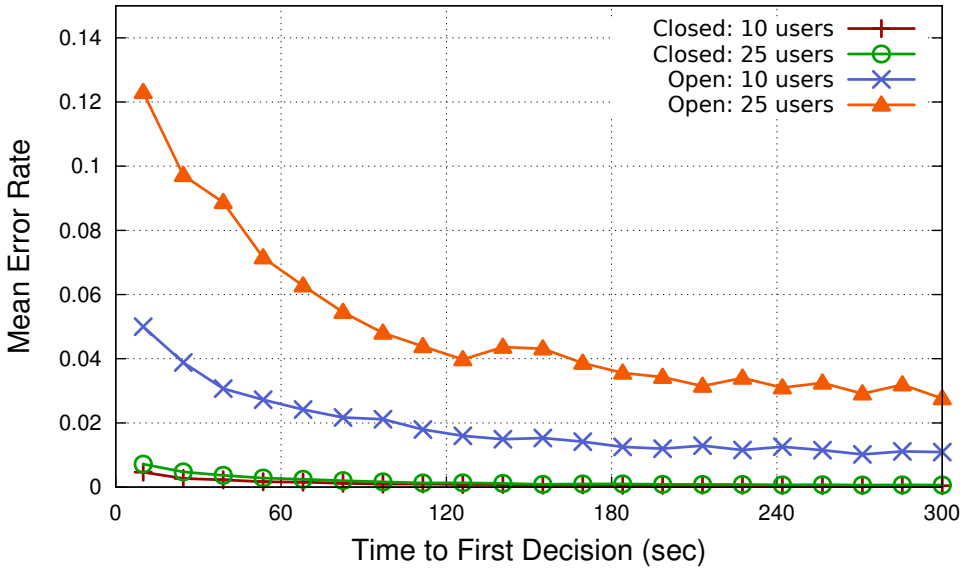

Figure 3.14: Comparing the performance of the fusion when all tested users are part of the training versus when only half of the tested users are part of the training. The standard deviation of each data point is small, with the coefficient of variation less than 0.5 for each point.

4. Authentication on Mobile Devices

4.1 Overview

According to a 2013 Pew Internet Project study of 2076 people [20], 91% of American adults own a cellphone. Increasingly, people are using their phones to access and store sensitive data. The same study found that 81% of cellphone owners use their mobile device for texting, 52% use it for email, 49% use it for maps (enabling location services), and 29% use it for online banking. And yet, securing the data is often not taken seriously because of an inaccurate estimation of risk as discussed in [22]. In particular, several studies have shown that a large percentage of smartphone owners do not lock their phone: 57% in [30], 33% in [64], 39% in [22], and 48% in this study.

Active authentication is an approach of monitoring the behavioral biometric characteristics of a user's interaction with the device for the purpose of securing the phone when the point-of-entry locking mechanism fails or is absent. In recent years, continuous authentication has been explored extensively on desktop computers, based either on a single biometric modality like mouse movement [56] or a fusion of multiple modalities like keyboard dynamics, mouse movement, web browsing, and stylometry [24]. Unlike physical biometric devices like fingerprint scanners or iris scanners, these systems rely on computer interface hardware like the keyboard and mouse that are already commonly available with most computers.

In this section, we consider the problem of active authentication on mobile devices, where the variety of available classifier data is much greater than on the desktop, but so is the variety of behavioral profiles, device form factors, and environments in which the device is used. We study four representative modalities of stylometry (text analysis), application usage patterns, web browsing behavior, and physical location of the device. In the remainder of the thesis these four modalities will be referred to as TEXT, APP, WEB, and LOCATION, respectively. We consider the trade-off between intruder detection time and detection error as measured by false accept rate (FAR) and false reject rate (FRR). The analysis is performed on a dataset collected by the authors of 200 subjects using their personal Android mobile device for a period of at least 30 days. To the best of our knowledge, this dataset is the first of its kind studied in active authentication literature, due to its large size [19], the duration of tracked activity [45], and the absence of restrictions on usage patterns and on the form factor of the mobile device. The geographical colocation of the participants, in particular,

makes the dataset a good representation of an environment such as a closed-world organization where the unauthorized user of a particular device will most likely come from inside the organization.

We propose to use decision fusion in order to asynchronously integrate the four modalities and make serial authentication decisions. While we consider here a specific set of binary classifiers, the strength of our decision-level approach is that additional classifiers can be added without having to change the basic fusion rule. Moreover, it is easy to evaluate the marginal improvement of any added classifier to the overall performance of the system. We evaluate the multimodal continuous authentication system by characterizing the error rates of local classifier decisions, fused global decisions, and the contribution of each local classifier to the fused decision. The novel aspects of our work include the scope of the dataset, the particular portfolio of behavioral biometrics in the context of mobile devices, and the extent of temporal performance analysis.

4.2 Dataset

The dataset used in this work contains behavioral biometrics data for 200 subjects. The collection of the data was carried out by the authors over a period of 5 months. The requirements of the study were that each subject was a student or employee of Drexel University and was an owner and an active user of an Android smartphone or tablet. The number of subjects with each major Android version and associated API level are listed in Table 4.1. Nexus 5 was the most popular device with 10 subjects using it. Samsung Galaxy S5 was the second most popular device with 6 subjects using it.

Android Version	**API Level**	**Subjects**
4.4	19	143
4.1	16	16
4.3	18	15
4.2	17	9
4.0.4	15	5
2.3.6	10	4
4.0.3	15	3
2.3.5	10	3
2.2	8	2

Table 4.1: The Android version and API level of the 200 devices that were part of the study.

A tracking application was installed on each subject's device and operated for a period of at least

30 days until the subject came in to approve the collected data and get the tracking application uninstalled from their device. The following data modalities were tracked with 1-second resolution:

- Text typed via soft keyboard.
- Apps visited.
- Websites visited.
- Location (based on GPS or WiFi).

The key characteristics of this dataset are its large size (200 users), the duration of tracked activity (30+ days), and the geographical colocation of its participants in the Philadelphia area. Moreover, we did not place any restrictions on usage patterns, on the type of Android device, and on the Android OS version (see Table 4.1).

There were several challenges encountered in the collection of the data. The biggest problem was battery drain. Due to the long duration of the study, we could not enable modalities whose tracking proved to be significantly draining of battery power. These modalities include front-facing video for eye tracking and face recognition, gyroscope, accelerometer, and touch gestures. Moreover, we had to reduce GPS sampling frequency to once per minute on most of the devices.

Event	**Frequency**
Text	23,254,478
App	927,433
Web	210,322
Location	143,875

Table 4.2: The number of events in the dataset associated with each of the four modalities considered in this thesis. A TEXT event refers to a single character entered on the soft keyboard. An APP events refers to a new app receiving focus. A WEB event refers to a new url entered in the url box. A LOCATION event refers to a new sample of the device location either from GPS or WiFi.

Table 4.2 shows statistics on each of the four investigated modalities in the corpus. The table contains data aggregated over all 200 users. The "frequency" here is a count of the number of instances of an action associated with that modality. As stated previously, the four modalities will be referred to as TEXT, APP, WEB, and "location." For TEXT, the action is a single keystroke on the soft keyboard. For APP, the action is opening or bringing focus to a new app. For WEB, the action

is visiting a new website. For LOCATION, no explicitly action is taken by the user. Rather, location is sampled regularly at intervals of 1 minute when GPS is enabled. As Table 4.2 suggests, TEXT events fire 1-2 orders of magnitude more frequently than the other three.

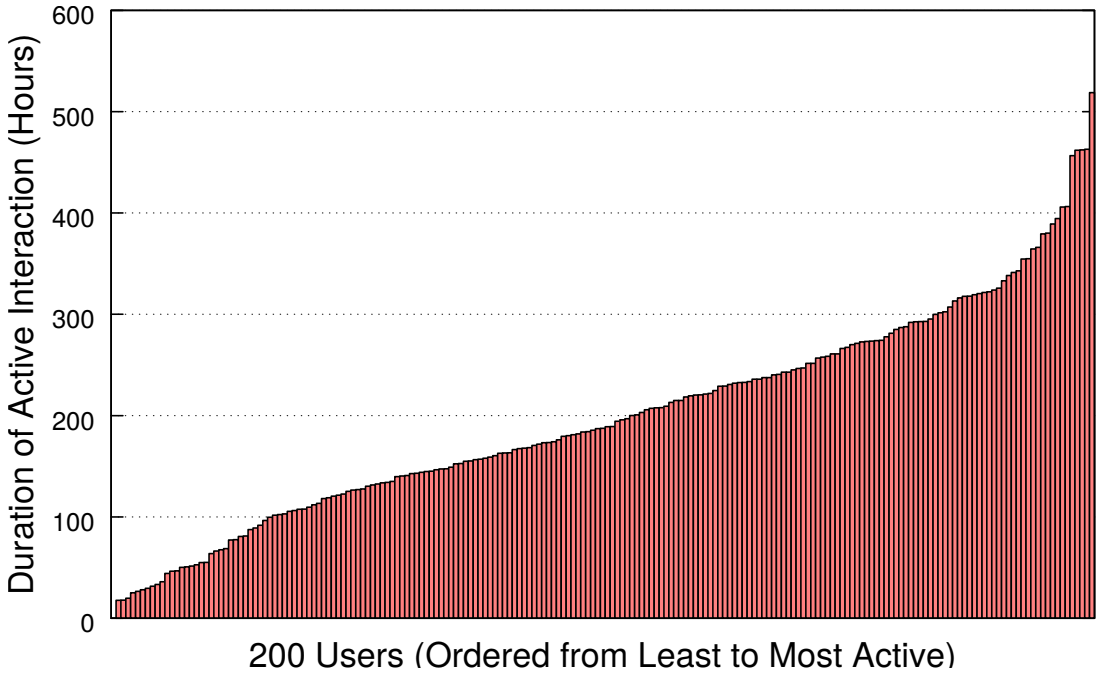

Figure 4.1: The duration of time (in hours) that each of the 200 users actively interacted with their device..

The data for each user is processed to remove idle periods when the device is not active. The threshold for what is considered an idle period is 5 minutes. For example, if the time between event A and event B is 20 minutes, with no other events in between, this 20 minutes is compressed down to 5 minutes. The date and time of the event are not changed but the timestamp used in dividing the dataset for training and testing (see §4.4.1) is updated to reflect the new time between event A and event B. This compression of idle times is performed in order to regularize periods of activity for cross validation that utilizes time-based windows as described in §4.4.1. The resulting compressed timestamps are referred to as "active interaction". Fig. 4.1 shows the duration (in hours) of active interaction for each of the 200 users ordered from least to most active.

Table 4.3 shows three top-20 lists: (1) the top-20 apps based on the amount of text that was

(a)

App Name	Keys Per App
com.android.sms	5,617,297
com.android.mms	5,552,079
com.whatsapp	4,055,622
com.facebook.orca	1,252,456
com.google.android.talk	1,147,295
com.infraware.polarisviewer4	990,319
com.android.chrome	417,165
com.facebook.katana	405,267
com.snapchat.android	377,840
com.google.android.gm	271,570
com.htc.sense.mms	238,300
com.tencent.mm	221,461
com.motorola.messaging	203,649
com.android.calculator2	167,435
com.verizon.messaging.vzmsgs	137,339
com.groupme.android	134,896
com.handcent.nextsms	123,065
com.jb.gosms	118,316
com.sonyericsson.conversations	114,219
com.twitter.android	92,605

(b)

App Name	Visits
TouchWiz home	101,151
WhatsApp	64,038
Messaging	60,015
Launcher	39,113
Facebook	38,591
Google Search	32,947
Chrome	32,032
Snapchat	23,481
System UI	22,772
Phone	19,396
Gmail	19,329
Messages	19,154
Contacts	18,668
Hangouts	17,209
Home	16,775
HTC Sense	16,325
YouTube	14,552
Xperia Home	13,639
Instagram	13,146
Settings	12,675

(c)

Website Domain	Visits
www.google.com	19,004
m.facebook.com	9,300
www.reddit.com	4,348
forums.huaren.us	3,093
learn.dcollege.net	2,133
en.m.wikipedia.org	1,825
mail.drexel.edu	1,520
one.drexel.edu	1,472
login.drexel.edu	1,462
likes.com	1,361
mail.google.com	1,292
i.imgur.com	1,132
www.amazon.com	1,079
netcontrol.irt.drexel.edu	1,049
www.facebook.com	903
banner.drexel.edu	902
m.hupu.com	824
t.co	801
duapp2.drexel.edu	786
m.ign.com	725

Table 4.3: Top 20 apps ordered by text entry and visit frequency and top 20 websites ordered by visit frequency. These tables are provided to give insight into the structure and content of the dataset.

typed inside each app, (2) the top-20 apps based on the number of times they received focused, and (3) the top-20 website domains based on the number of times a website associated with that domain was visited. These are aggregate measures across the dataset intended to provide an intuition about its structure and content, but the top-20 list is the same as that used for the the classifier model based on the WEB and APP features in §4.3.

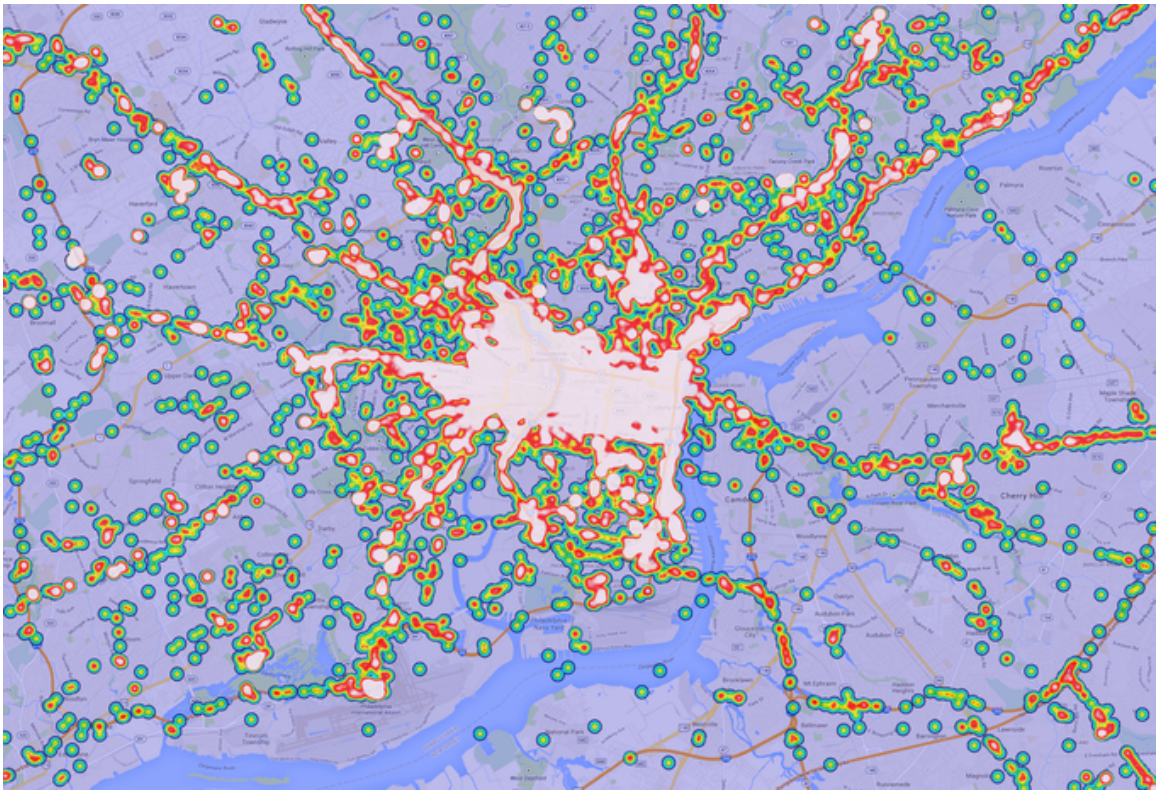

Figure 4.2: An aggregate heatmap showing a selection from the dataset of GPS locations in the Philadelphia area.

Fig. 4.2 shows a heat map visualization of a selection from the dataset of GPS locations in the Philadelphia area. The subjects in the study resided in Philadelphia but traveled all over United States and the world. There are two key characteristics of the GPS location data. First, it is relatively unique to each individual even for people living in the same area of a city. Second, outside of occasional travel, it does not vary significantly from day to day. Human beings are creatures of habit, and in as much as location is a measure of habit, this idea is confirmed by the location data of the majority of the subjects in the study.

4.3 Classification and Decision Fusion

4.3.1 Features and Classifiers

The four distinct biometric modalities considered in our analysis are (1) text entered via soft keyboard, (2) applications used, (3) websites visited, and (4) physical location of the device as determined from GPS (when outdoors) or WiFi (when indoors). We refer to these four modalities as TEXT, APP, WEB, and LOCATION, respectively. In this section we discuss the features that were extracted from the raw data of each modality, and the classifiers that were used to map these features into binary decision space.

A binary classifier is constructed for each of the 200 users and 4 modalities. In total, there are 800 classifiers, each producing either a probability that a user is valid $P(H_1)$ (or a binary decision of 0 (invalid) or 1 (valid). The first class (H_1) for each classifier is trained on the valid user's data and the second class (H_0) is trained on the other 199 users' data. The training process is described in more detail in §4.4.1. For APP, WEB, and LOCATION, the classifier takes a single instance of the event and produces a probability. For multiple events of the same modality, the set of probabilities is fused across time using maximum likelihood:

$$H^* = \underset{i \in \{0,1\}}{\operatorname{argmax}} \prod_{x_t \in \Omega} P(x_t | H_i), \tag{4.1}$$

where $\Omega = \{x_t | T_{\text{current}} - T(x_t) \leq \omega\}$, ω is a fixed window size in seconds, $T(x_t)$ is the timestamp of event x_t, and T_{current} is the current timestamp. The process of fusing classifier scores across time is illustrated in Fig. 4.3.

Text

As Table 4.2a indicates, the apps into which text was entered on mobile devices varied, but the activity in majority of the cases was communication via SMS, MMS, WhatsApp, Facebook, Google Hangouts, and other chat apps. Therefore, TEXT events fired in short bursts. The tracking application captured the keys that were touched on the keyboard and not the autocorrected result. Therefore, the majority of the typed messages had a lot of misspellings and words that were erased in the final submitted message. In the case of SMS, we also were able to record the submitted result. For example, an SMS text that was submitted as "`Sorry couldn't call back.`" had associated with it the following recorded keystrokes: "`Sprry coyld cpuldn't vsll back.`" Classification

based on the actual typed keys in principle is a better representation of the person's linguistic style. It captures unique typing idiosyncrasies that autocorrect can conceal. As discussed in §2, we implemented a one-feature n-gram classifier from [15] that has been shown to work well on short messages. It works by analyzing the presence or absence of n-grams with respect to the training set.

App and Web

The APP and WEB classifier models we construct are identical in their structure. For the APP modality we use the app name as the unique identifier and count the number of times a user visits each app in the training set. For the WEB modality we use the domain of the URL as the unique identifier and count the number of times a user visits each domain in the training set. Note that, for example, "m.facebook.com" is a considered a different domain than "www.facebook.com" because the subdomain is different. In this section we refer to the app name and the web domain as an "entity". Table 4.2b and Table 4.2c show the top entities aggregated across all 200 users for APP and WEB respectively.

For each user, the classification model for the valid class is constructed by determining the top 20 entities visited by that user in the training set. The quantity of visits is then normalized so that the 20 frequency values sum to 1. The classification model for the invalid class is constructed by counting the number of visit by the other 199 users to those same 20 domains, such that for each of those domains we now have a probability that a valid user visits it and an invalid user visits it. The evaluation for each user given the two empirical distributions is performed by the maximum likelihood product in (4.1). Entities that do not appear in the top 20 are considered outliers and are ignored in this classifier.

Location

Location is specified as a pair of values: latitude and longitude. Classification is performed using support vector machines (SVMs) [4] with the radial basis function (RBF) as the kernel function. The SVM produces a classification score for each pair of latitude and longitude. This score is calibrated to form a probability using Platt scaling [48] which requires an extra logistic regression on the SVM scores via an additional cross-validation on the training data. All of the code in this thesis is written by the authors except for the SVM classifier. Since the authentication system is written in C++, we used the Shark 3.0 machine learning library for the SVM implementation.

4.3.2 Decision Fusion

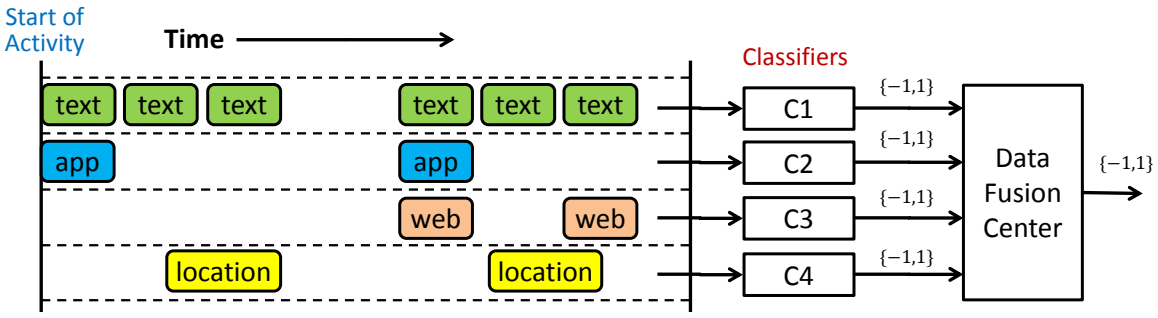

Figure 4.3: The fusion architecture across time and across classifiers. The TEXT, APP, WEB, and LOCATION boxes indicate a firing of a single event associated with each of those modalities. Multiple classifier scores from the same modality are fused via (4.1) to produce a single local binary decision. Local binary decisions from each of the four modalities are fused via (4.4) to produce a single global binary decision.

Decision fusion with distributed classifiers is described by Tenney and Sandell in [62] who studied a parallel decision architecture. As described in [36], the system comprises of n local detectors, each making a decision about a binary hypothesis (H_0, H_1), and a decision fusion center (DFC) that uses these local decisions $\{u_1, u_2, ..., u_n\}$ for a global decision about the hypothesis. The i^{th} detector collects K observations before it makes its decision, u_i. The decision is $u_i = 1$ if the detector decides in favor of H_1 and $u_i = -1$ if it decides in favor of H_0. The DFC collects the n decisions of the local detectors and uses them in order to decide in favor of $H_0(u = -1)$ or in favor of $H_1(u = 1)$. Tenney and Sandell [62] and Reibman and Nolte [53] studied the design of the local detectors and the DFC with respect to a Bayesian cost, assuming the observations are independent conditioned on the hypothesis. The ensuing formulation derived the local and DFC decision rules to be used by the system components for optimizing the system-wide cost. The resulting design requires the use of likelihood ratio tests by the decision makers (local detectors and DFC) in the system. However the thresholds used by these tests require the solution of a set of nonlinear coupled differential equations. In other words, the design of the local decision makers and the DFC are co-dependent. In most scenarios the resulting complexity renders the quest for an optimal design impractical.

Chair and Varshney in [17] developed the optimal fusion rule when the local detectors are fixed and local observations are statistically independent conditioned on the hypothesis. Data Fusion

Center is optimal given the performance characteristics of the local fixed decision makers. The result is a suboptimal (since local detectors are fixed) but computationally efficient and scalable design. In this study we use the Chair-Varshney formulation. The parallel distributed fusion scheme (see Fig. 4.3) allows each classifier to observe an event, minimize the local risk and make a local decision over the set of hypothesis, based on only its own observations. Each classifier sends out a decision of the form:

$$u_i = \begin{cases} 1, & \text{if } H_1 \text{ is decided} \\ -1, & \text{if } H_0 \text{ is decided} \end{cases} \tag{4.2}$$

The fusion center combines these local decisions by minimizing the global Bayes' risk. The optimum decision rule performs the following likelihood ratio test

$$\frac{P(u_1, ..., u_n|H_1)}{P(u_1, ..., u_n|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1} = \tau \tag{4.3}$$

where the a priori probabilities of the binary hypotheses H_1 and H_0 are P_1 and P_0 respectively. In this case the general fusion rule proposed in [17] is

$$f(u_1, ..., u_n) = \begin{cases} 1, & \text{if } a_0 + \sum_{i=0}^{n} a_i u_i > 0 \\ -1, & \text{otherwise} \end{cases} \tag{4.4}$$

with P_i^M, P_i^F representing the *False Rejection Rate* (FRR) and *False Acceptance Rate* (FAR) of the i^{th} classifier respectively. The optimum weights minimizing the global probability of error are given by

$$a_0 = \log \frac{P_1}{P_0} \tag{4.5}$$

$$a_i = \begin{cases} \log \frac{1-P_i^M}{P_i^F}, & \text{if } u_i = 1 \\ \log \frac{1-P_i^F}{P_i^M}, & \text{if } u_i = -1 \end{cases} \tag{4.6}$$

The threshold in (4.3) requires knowledge of the a priori probabilities of the hypotheses. In practice, these probabilities are not available, and the threshold τ is determined using different considerations such as fixing the probability of false alarm or false rejection as is done in §4.4.3.

4.4 Results

4.4.1 Training, Characterization, Testing

The data of each of the 200 users' active interaction with the mobile device was divided into 5 equal-size folds (each containing 20% time span of the full set). We performed training of each classifier on the first three folds (60%). We then tested their performance on the fourth fold. This phase is referred to as "characterization", because its sole purpose is to form estimates of FAR and FRR for use by the fusion algorithm. We then tested the performance of the classifiers, individually and as part of the fusion system, on the fifth fold. This phase is referred to as "testing" since this is the part that is used for evaluation the performance of the individual classifiers and the fusion system. The three phases of training, characterization, and testing as they relate to the data folds are shown in Fig. 4.4.

- Training on folds 1, 2, 3.
  Characterization on fold 4.
  Testing on fold 5.
- Training on folds 2, 3, 4.
  Characterization on fold 5.
  Testing on fold 1.
- Training on folds 3, 4, 5.
  Characterization on fold 1.
  Testing on fold 2.
- Training on folds 4, 5, 1.
  Characterization on fold 2.
  Testing on fold 3.
- Training on folds 5, 1, 2.
  Characterization on fold 3.
  Testing on fold 4.

The common evaluation method used with each classifier for data fusion was measuring the averaged error rates across five experiments; In each experiment, data of 3 folds was taken for training, 1 fold for characterization, and 1 for testing. The FAR and FRR computed during characterization

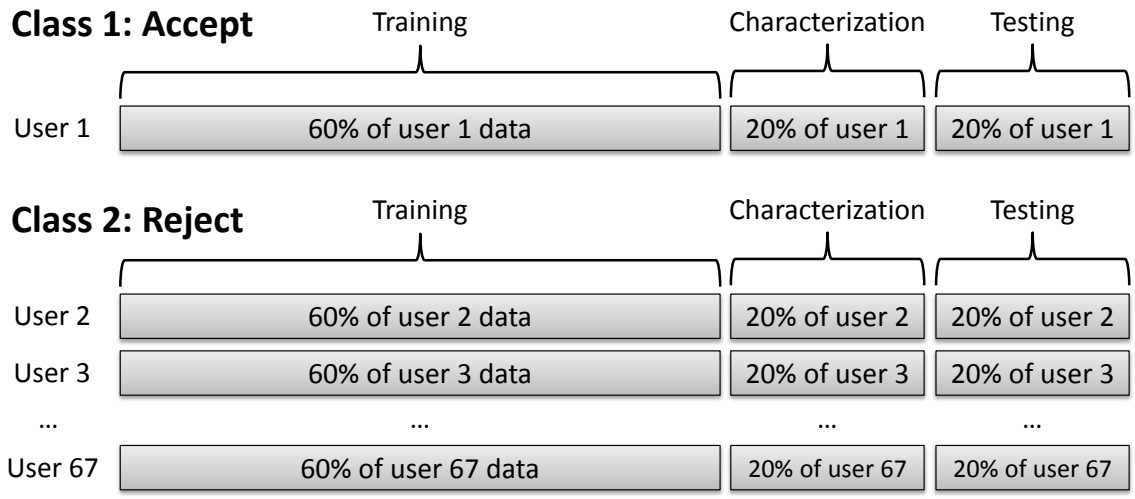

Figure 4.4: The three phases of processing the data to determine the individual performance of each classifiers and the performance of the fusion system that combines some subset of these classifiers.

were taken as input for the fusion system as a measurement of the expected performance of the classifiers. Therefore each experiment consisted of three phases: 1) train the classifier(s) using the training set, 2) determine FAR and FRR based on the training set, and 3) classify the windows in the test set.

4.4.2 Performance: Individual Classifiers

The conflicting objectives of an active authentication system are of response-time and performance. The less the system waits before making an authentication decision, the higher the expected rate of error. As more behavioral biometric data trickles in, the system can, on average, make a classification decision with greater certainty.

This pattern of decreased error rates with an increased decision window can be observed in Fig. 4.5 that shows (for 10 different time windows) the FAR and FRR of the 4 classifiers averaged over the 200 users with the error bars indicating the standard deviation. The "testing fold" (see §4.4.1) is used for computing these error rates. The "characterization fold" does not affect these results, but is used only for FAR/FRR estimation required by the decision fusion center in §4.4.3.

The "time before decision" is the time between the first event indicating activity and the first decision produced by the fusion system. This metric can be thought of as "decision window size". Events older than the time range covered by the time-window are disregarded in the classification. If no event associated with the modality under consideration fires in a specific time window, no error

is added to the average.

Event	Firing Rate (per hour)
Text	557.8
App	23.2
Web	5.6
Location	3.5

Table 4.4: The rates at which an event associated with each modality "fires" per hour. On average, GPS location is provided only 3.5 times an hour.

There are two notable observations about the FAR/FRR plots in Fig. 4.5. First, the location modality provides the lowest error rates even though on average across the dataset it fires only 3.5 times an hour as shown in Table 4.4. This means that classification on a single GPS coordinate is sufficient to correctly verify the user with an FAR of under 0.1 and an FRR of under 0.05. Second, the text modality converges to an FAR of 0.16 and an FRR of 0.11 after 30 minutes which is one of the worse performers of the four modalities, even though it fires 557.8 times an hour on average. At the 30 minute mark, that firing rate equates to an average text block size of 279 characters. An FAR/FRR of 0.16/0.11 with 279 characters blocks improves on the error rates achieved in [15] with 500 character blocks which in turn improved on the errors rates achieved in prior work for blocks of small text (see [15] for a full reference list on short-text stylometric analysis).

4.4.3 Performance: Decision Fusion

The events associated with each of the 4 modalities fire at very different rates as shown in Table 4.4. Moreover, text events fire in bursts, while the location events fire at regularly spaced intervals when GPS signal is available. The app and web events fire at varying degrees of burstiness depending on the user. Fig. 4.6 shows the distribution of the number of events that fire within each of the time windows. An important takeaway from these distributions is that most events come in bursts followed by periods of inactivity. This results in the counterintuitive fact that the 1 minute, 10 minute, and 30 minute windows have a similar distribution on the number of events that fire within them. This is why the decrease in error rates attained from waiting longer for a decision is not as significant as might be expected.

Asynchronous fusion of classification of events from each of the four modalities is robust to

the irregular rates at which events fire. The decision fusion rule in (4.4) utilizes all the available biometric data, weighing each classifier according to its prior performance. Fig. 4.7 shows the receiver operating characteristic (ROC) curve trading off between FAR and FRR by varying the threshold parameter τ in (4.3).

As the size of the decision window increases, the performance of the fusion system improves, dropping from an equal error rate (EER) of 0.05 using the 1 minute window to below 0.01 EER using the 30 minute window.

4.4.4 Contribution of Local Classifiers to Global Decision

The performance of the fusion system that utilizes all four modalities of TEXT, APP, WEB, and LOCATION is described in the previous section. Besides this, we are able to use the fusion system to characterize the contribution of each of the local classifiers to the global decision. This is the central question we consider in the thesis: what biometric modality is most helpful in verifying a person's identity under a constraint of a specific time window before the verification decision must be made? We measure the contribution C_i of each of the four classifiers by evaluating the performance of the system with and without the classifier, and computing the contribution by:

$$C_i = \frac{E_i - E}{E} \tag{4.7}$$

where E is the error rate computed by averaging FAR and FRR of the fusion system using the full portfolio of 4 classifiers, E_i is the error rate of the fusion system using all but the i-th classifier, and C_i is the relative contribution of the i-th classifier as shown in Fig. 4.8. We consider the contribution of each classifier under three time windows of 1 minute, 10 minutes, and 30 minutes. Location contributes the most in all three cases, with the second biggest contributor being web browsing. Text contributes the least for the small window of 1 minute, but improve for the large windows. App usage is the least predictable contributor.

4.5 Conclusion

In this work, we proposed a parallel binary decision-level fusion architecture for classifiers based on four biometric modalities: text, application usage, web browsing, and location. Using this fusion method we addressed the problem of active authentication and characterized its performance on a real-world dataset of 200 subjects, each using their personal Android mobile device for a period of

at least 30 days. The authentication system achieved an equal error rate (ERR) of 0.05 (5%) after 1 minute of user interaction with the device, and an EER of 0.01 (1%) after 30 minutes. We showed the performance of each individual classifier and its contribution to the fused global decision. The location-based classifier, while having the lowest firing rate, contributes the most to the performance of the fusion system.

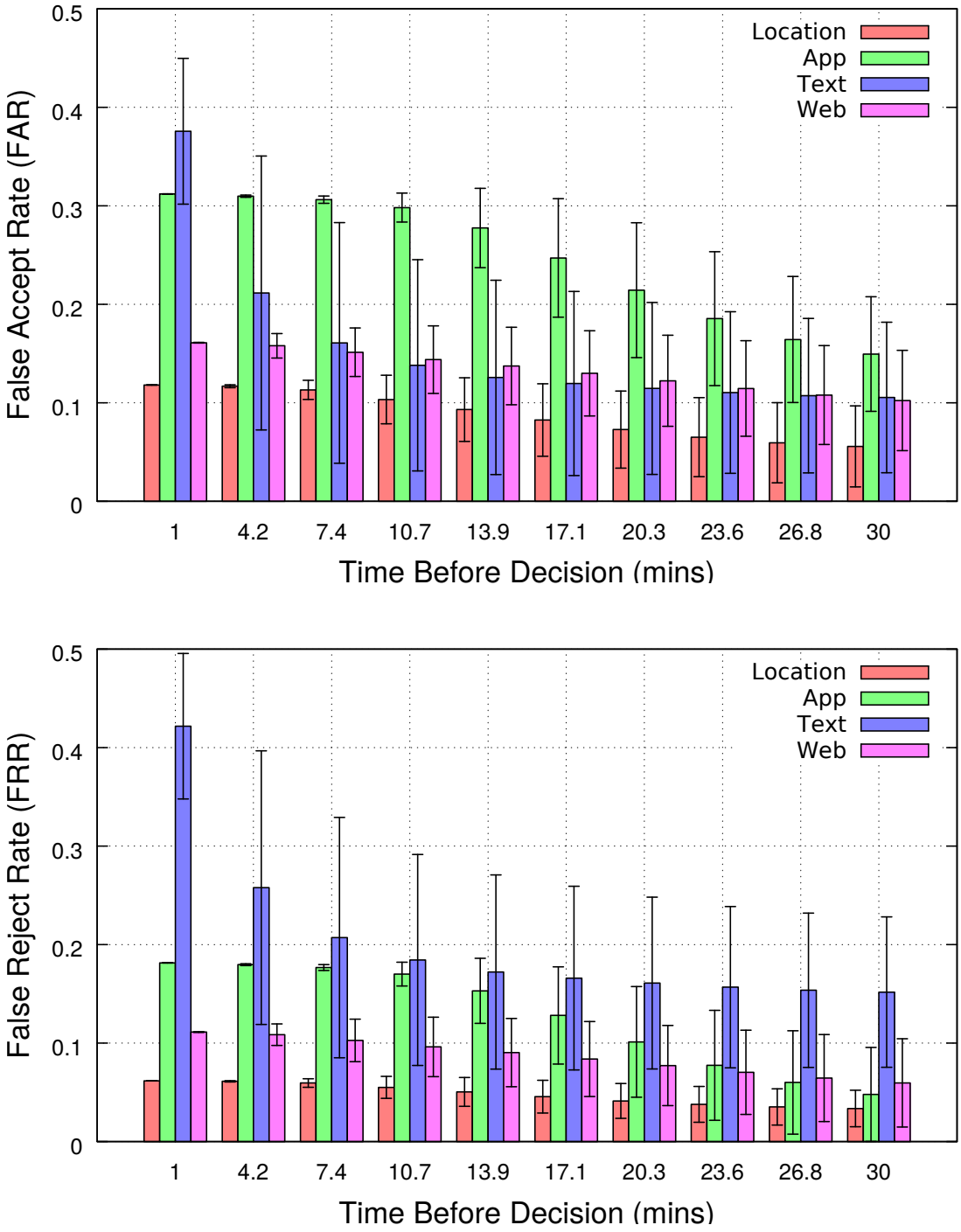

Figure 4.5: FAR and FRR performance of the individual classifiers associated with each of the four modalities. Each bar represent the average error rate for a given module and time window. Each of the 200 users has 2 classifiers for each modality, so each bar provides a value that was averaged over 200 individual error rates. The error bar indicate the standard deviation across these 200 values.

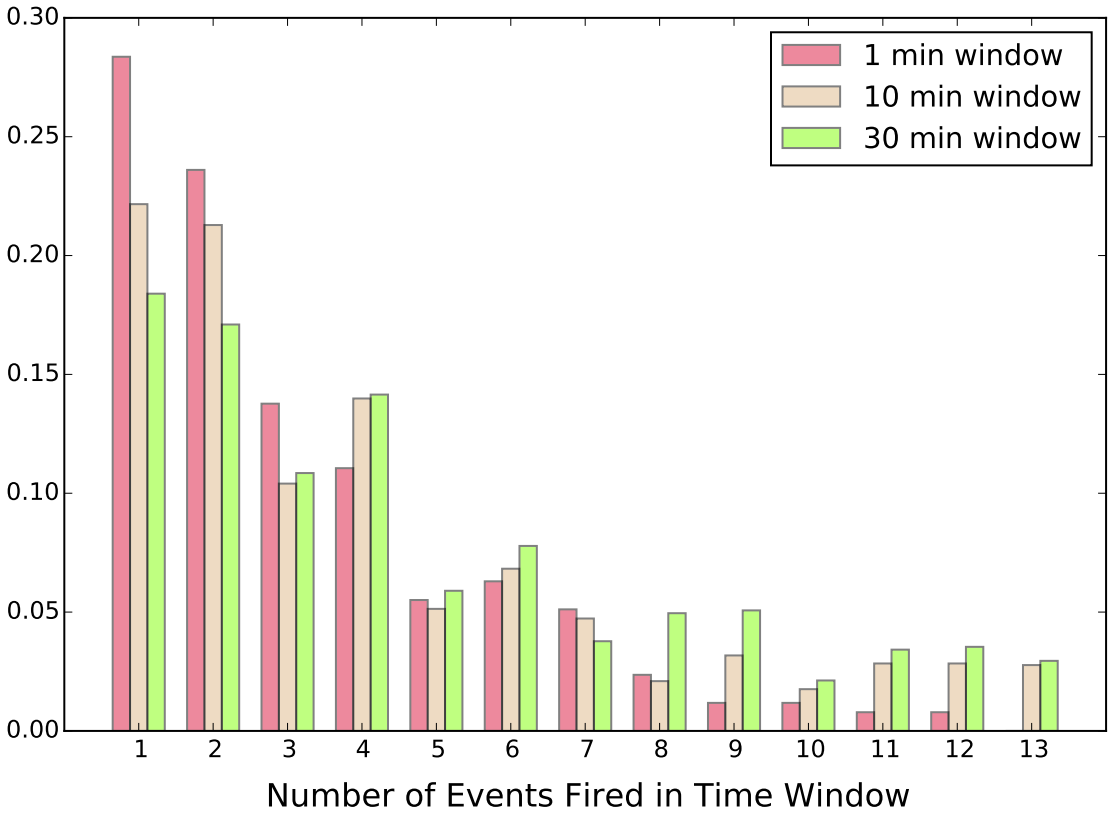

Figure 4.6: The distribution of the number of events that fire within a given time window. This is a long tail distribution as non-zero probabilities of event frequencies above 13 extend to over 100. These outliers are excluded from this histogram plot in order to highlight the high-probability frequencies. Time windows in which no events fire are not included in this plot.

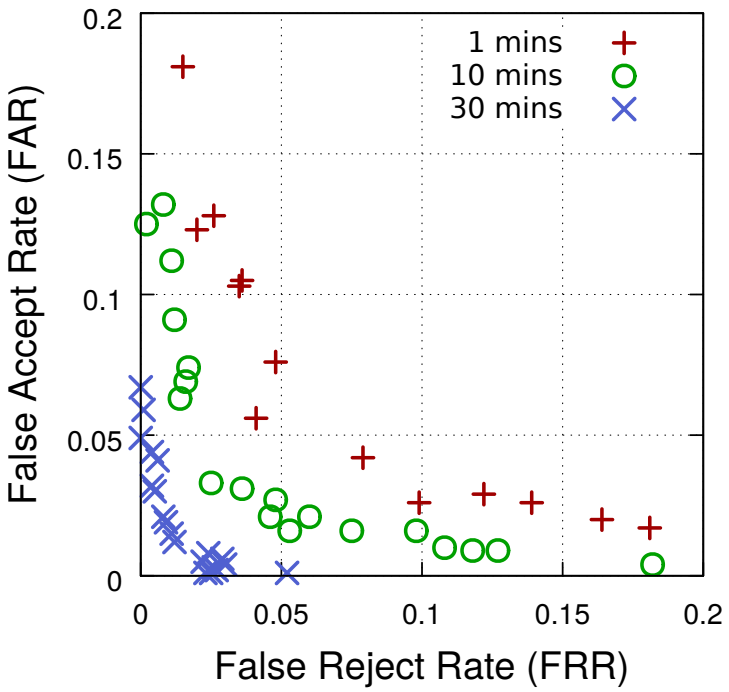

Figure 4.7: The performance of the fusion system with 4 classifiers on the 200 subject dataset. The ROC curve shows the tradeoff between FAR and FRR achieved by varying the threshold parameter a_0 in (4.4).

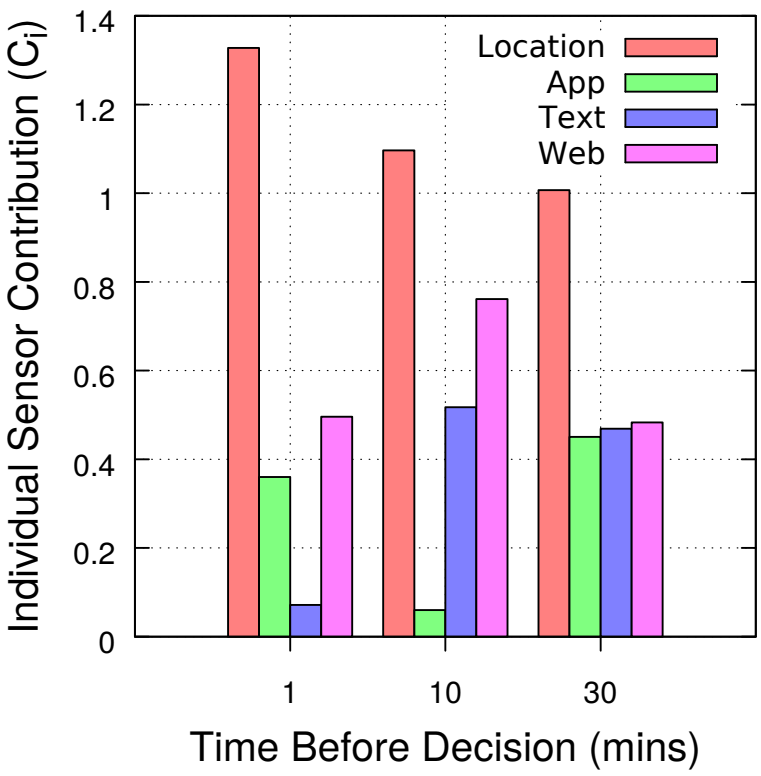

Figure 4.8: Relative contribution of each of the 4 classifiers computed according to (4.7).

5. Conclusion

We propose a parallel binary detection decision fusion architecture for a representative collection of behavioral biometric classifiers: keystroke dynamics, mouse movement, and stylometry. Using this fusion method we address the problem of active authentication and characterize its performance on a dataset from a real-world office environment. We consider several applications for this authentication system, with a particular focus on secure distributed communication, because the training of the classifiers requires biometric data from multiple users on the network: the "legitimate" user at each computer under inspection and the "illegitimate" users who may try to use a computer they are not authorized to access.

The application of the Chair-Varshney fusion algorithm to the problem of multi-modal authentication and the use of high-level classifiers based on stylometry are novel in the continuous authentication context, and show promising performance in terms of low false acceptance rate (FAR) and low false rejection rate (FRR). We observe the tradeoff between detection time and error rate, and show that error rates of less than 0.01 can be achieved in under 60 seconds of active computer use. We also demonstrate that the system is robust to partial spoofing of the classifiers.

We also evaluated the fusion system on a mobile dataset of 200 subjects, each using their personal Android mobile device for a period of at least 30 days. The authentication system achieved an equal error rate (ERR) of 0.05 (5%) after 1 minute of user interaction with the device, and an EER of 0.01 (1%) after 30 minutes. We showed the performance of each individual classifier and its contribution to the fused global decision. The location-based classifier, while having the lowest firing rate, contributes the most to the performance of the fusion system.

We consider several directions for future work. First, we aim to examine a wider variety and combination of metrics, based on keystrokes, mouse events and any application data that can be monitored and collected (e.g. web browsing behavior). These may include the classifiers presented in this thesis, in different application scopes. For instance, since different domains may require specific features to capture unique characteristics of those domains, we can use several custom-made stylometric metrics for word processors, mail clients, short-message domains (e.g. instant messaging, Twitter [44], Facebook) etc. In addition, novel metrics can be developed that use a combination of input types. For instance, it may be useful to examine periods of dual mouse and keyboard usage (e.g. selecting text with the mouse and using keyboard shortcuts to rearrange it) and parameterize

the synchronization between them.

Each of the classifiers considered have a set of parameters that control their behavior and resulting performance. We intend to further explore the effect of changing these parameters, e.g., size of the training set. An important variable to examine is time-based vs. data-based windows; as opposed to time-based windows (used for this thesis), data-based windows mean that each classifier generates a decision when enough of the data it is based on is aggregated, resulting with asynchronous decision making. On top of this approach, parameters of individual classifier data thresholds and maximum window time can be set and tested compared to fixed, synchronous time-based windows.

Moreover, the current implementation of the system is designed for post-mortem analysis and classification, based on per-user decision comma-separated values (CSV) files generated by each of the classifiers. For future implementation improvements we propose upgrading to live data collection and analysis, as expected to perform in real settings. We propose using relational databases to store: 1) directly-collected metrics, like mouse events, keystrokes, web browsing statistics, clipboard content and any potential usable application data, and 2) decisions from any participating classifiers generated on-the-fly. Since collecting raw data involves security and privacy risks, it may be considered to collect only parsed, extracted vectors of information generated by the participating classifiers. For instance, instead of storing the sequence of keystrokes for a particular window, only the vector of statistics extracted from that sequence will be stored, for each classifier that uses this information. The disadvantages are that post-mortem analysis cannot be applied using potential new classifiers/configurations, as the raw data will not be available. The clear advantage is that the sensitivity of the stored information is reduced. In either case, all databases should be stored in a secure encrypted storage, and managed carefully when processed to discourage information leakage. The implementation improvement suggested above can also allow convenient remote access by centralized authentication systems, configured with different sets of classifiers.

Finally, the usability of the system is determined by its ability to detect intruders, but more importantly, raise false alarms as little as possible. Surely a system that prompts the user for password frequently due to misclassification as an intruder has severe usability issues, let alone annoying. Therefore adding support for user-defined target FRR and FAR thresholds (given classifiers that can hold up to to them) is an important setting of the system, to allow the ability to determine the minimum performance the system is expected to work with.

Bibliography

[1] A. Abbasi and H. Chen. Identification and comparison of extremist-group web forum messages using authorship analysis. *IEEE Intelligent Systems*, 20(5):67–75, 2005.

[2] Ahmed Abbasi and Hsinchun Chen. Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace. *ACM Transactions on Information Systems (TOIS)*, 26(2):7, 2008.

[3] Ahmed Abbasi and Hsinchun Chen. Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace. *ACM Trans. Inf. Syst.*, 26(2):7:1–7:29, April 2008.

[4] Shigeo Abe. *Support vector machines for pattern classification*. Springer, 2010.

[5] Myriam Abramson and David W Aha. User authentication from web browsing behavior. In *FLAIRS Conference*, 2013.

[6] A.A.E. Ahmed and I. Traore. A new biometric technology based on mouse dynamics. *Dependable and Secure Computing, IEEE Transactions on*, 4(3):165 –179, july-sept. 2007.

[7] K.M. Ali and M.J. Pazzani. *On the link between error correlation and error reduction in decision tree ensembles*. Citeseer, 1995.

[8] Shlomo Argamon, Moshe Koppel, James W. Pennebaker, and Jonathan Schler. Automatically profiling the author of an anonymous text. *CACM*, 52(2):119–123, February 2009.

[9] Kyle O. Bailey, James S. Okolica, and Gilbert L. Peterson. User identification and authentication using multi-modal behavioral biometrics. *Computers and Security*, 43(0):77 – 89, 2014.

[10] Ned Bakelman, John V. Monaco, Sung-Hyuk Cha, and Charles C. Tappert. Continual keystroke biometric authentication on short bursts of keyboard input. In *Proceedings of Student-Faculty Research Day, CSIS, Pace University*, 2012.

[11] N. Bartlow and B. Cukic. Evaluating the reliability of credential hardening through keystroke dynamics. In *Software Reliability Engineering, 2006. ISSRE'06. 17th International Symposium on*, pages 117–126. IEEE, 2006.

[12] Francesco Bergadano, Daniele Gunetti, and Claudia Picardi. User authentication through keystroke dynamics. *ACM Trans. Inf. Syst. Secur.*, 5(4):367–397, November 2002.

[13] Jose Nilo G. Binongo. Who wrote the 15th book of Oz? an application of multivariate analysis to authorship attribution. *Chance*, 16(2):9–17, 2003.

[14] SA Bleha, J. Knopp, and MS Obaidat. Performance of the perceptron algorithm for the classification of computer users. In *Proceedings of the 1992 ACM/SIGAPP symposium on Applied computing: technological challenges of the 1990's*, pages 863–866. ACM, 1992.

[15] Marcelo Luiz Brocardo, Issa Traore, Sherif Saad, and Isaac Woungang. Authorship verification for short messages using stylometry. In *Computer, Information and Telecommunication Systems (CITS), 2013 International Conference on*, pages 1–6. IEEE, 2013.

[16] Marcelo Luiz Brocardo, Issa Traore, and Isaac Woungang. Toward a framework for continuous authentication using stylometry. In *Advanced Information Networking and Applications (AINA), 2014 IEEE 28th International Conference on*, pages 106–115. IEEE, 2014.

[17] Z. Chair and P.K. Varshney. Optimal data fusion in multiple sensor detection systems. *Aerospace and Electronic Systems, IEEE Transactions on*, AES-22(1):98 –101, jan. 1986.

[18] Ching-Han Chen and Ching-Yi Chen. Optimal fusion of multimodal biometric authentication using wavelet probabilistic neural network. In *Consumer Electronics (ISCE), 2013 IEEE 17th International Symposium on*, pages 55–56. IEEE, 2013.

[19] Mohammad Omar Derawi, Claudia Nickel, Patrick Bours, and Christoph Busch. Unobtrusive user-authentication on mobile phones using biometric gait recognition. In *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2010 Sixth International Conference on*, pages 306–311. IEEE, 2010.

[20] Maeve Duggan. Cell phone activities 2013. *Cell*, 2013.

[21] Nathan Eagle, Alex Sandy Pentland, and David Lazer. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences*, 106(36):15274–15278, 2009.

[22] Serge Egelman, Sakshi Jain, Rebecca S Portnoff, Kerwell Liao, Sunny Consolvo, and David Wagner. Are you ready to lock? In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 750–761. ACM, 2014.

[23] Clara Eusebi, Cosmin Gilga, Deepa John, and Andre Maisonave. A data mining study of mouse movement, stylometry, and keystroke biometric data. *Proc. CSIS Research Day, Pace Univ*, 2008.

[24] Alex Fridman, Ariel Stolerman, Sayandeep Acharya, Patrick Brennan, Patrick Juola, Rachel Greenstadt, and Moshe Kam. Decision fusion for multimodal active authentication. *IEEE IT Professional*, 15(4), July 2013.

[25] Alex Fridman, Ariel Stolerman, Sayandeep Acharya, Patrick Brennan, Patrick Juola, Rachel Greenstadt, and Moshe Kam. Multi-modal decision fusion for continuous authentication. *Computers and Electrical Engineering*, page Accepted, 2014.

[26] R. Giot, M. El-Abed, and C. Rosenberger. Keystroke dynamics authentication for collaborative systems. In *Collaborative Technologies and Systems, 2009. CTS'09. International Symposium on*, pages 172–179. IEEE, 2009.

[27] Christine Gray and Patrick Juola. Personality identification through on-line text analysis. In *Chicago Colloquium on Digital Humanities and Computer Science*, Chicago, IL, November 2011.

[28] Jeyanthi Hall, Michel Barbeau, and Evangelos Kranakis. Anomaly-based intrusion detection using mobility profiles of public transportation users. In *Wireless And Mobile Computing, Networking And Communications, 2005.(WiMob'2005), IEEE International Conference on*, volume 2, pages 17–24. IEEE, 2005.

[29] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November 2009.

[30] Marian Harbach, Emanuel von Zezschwitz, Andreas Fichtner, Alexander De Luca, and Matthew Smith. Itsa hard lock life: A field study of smartphone (un) locking behavior and risk perception. In *Symposium on Usable Privacy and Security (SOUPS)*, 2014.

[31] L.W. JAMES. Fundamentals of biometric authentication technologies. *International Journal of Image and Graphics*, 1(01):93–113, 2001.

[32] M. L. Jockers and D.M Witten. A comparative study of machine learning methods for authorship attribution. *LLC*, 25(2):215–23, 2010.

[33] A Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in neural information processing systems*, 14:841, 2002.

[34] Patrick Juola. Authorship attribution. *Foundations and Trends in Information Retrieval*, 1(3), 2006.

[35] Patrick Juola, Michael Ryan, and Michael Mehok. Geographically localizing tweets using stylometric analysis. In *Proceedings of the American Association of Corpus Linguistics 2011*, Atlanta, GA, 2011.

[36] Moshe Kam, Wei Chang, and Qiang Zhu. Hardware complexity of binary distributed detection systems with isolated local bayesian detectors. *IEEE Transactions on Systems Man and Cybernetics*, 21:565–571, 1991.

[37] M. Karnan, M. Akila, and N. Krishnaraj. Biometric personal authentication using keystroke dynamics: A review. *Applied Soft Computing*, 11(2):1565–1573, 2011.

[38] J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas. On combining classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(3):226–239, 1998.

[39] Moshe Koppel, Johnathan Schler, and K. Zigdon. Determining an author's native language by mining a text for errors (short paper). In *Proceedings of KDD*, Chicago,IL, August 2005.

[40] Moshe Koppel and Jonathan Schler. Ad-hoc authorship attribution competition approach outline. In Patrick Juola, editor, *Ad-hoc Authorship Attribution Contest*. ACH/ALLC 2004, 2004.

[41] Moshe Koppel and Jonathan Schler. Authorship verification as a one-class classification problem. In *Proceedings of the twenty-first international conference on Machine learning*, ICML '04, pages 62–, New York, NY, USA, 2004. ACM.

[42] Moshe Koppel, Jonathan Schler, and Shlomo Argamon. Computational methods in authorship attribution. *Journal of the American Society for Information Science and Technology*, 60(1):9–26, 2009.

[43] O. V. Kukushkina, A. A. Polikarpov, and D. V. Khmelev. Using literal and grammatical statistics for authorship attribution. *Problemy Peredachi Informatii*, 37(2):96–198, 2001. Translated in "Problems of Information Transmission," pp. 172–184.

[44] Robert Layton, Paul Watters, and Richard Dazeley. Authorship attribution for twitter in 140 characters or less. In *Proceedings of the 2010 Second Cybercrime and Trustworthy Computing Workshop*, CTC '10, pages 1–8, Washington, DC, USA, 2010. IEEE Computer Society.

[45] Fudong Li, Nathan Clarke, Maria Papadaki, and Paul Dowland. Active authentication for mobile devices utilising behaviour profiling. *International Journal of Information Security*, 13(3):229–244, 2014.

[46] Andrew W. E. McDonald, Sadia Afroz, Aylin Caliskan, Ariel Stolerman, and Rachel Greenstadt. Use fewer instances of the letter "i": Toward writing style anonymization. In *Lecture Notes in Computer Science*, volume 7384, pages 299–318. Springer, 2012.

[47] F. Mosteller and D. L. Wallace. *Inference and Disputed Authorship : The Federalist*. Addison-Wesley, Reading, MA, 1964.

[48] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632. ACM, 2005.

[49] M.S. Obaidat and B. Sadoun. Verification of computer users using keystroke dynamics. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 27(2):261 –269, apr 1997.

[50] T. Ord and SM Furnell. User authentication for keypad-based devices using keystroke analysis. In *Proceedings of the Second International Network Conference (INC-2000)*, pages 263–272, 2000.

[51] J. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schoelkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1998.

[52] M. Pusara and C.E. Brodley. User re-authentication via mouse movements. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, pages 1–8. ACM, 2004.

[53] Amy R. Reibman and L.W. Nolte. Optimal detection and performance of distributed sensor systems. *IEEE Transactions on Aerospace and Electronic Systems*, AES-23:24–30, 1987.

[54] Joseph Rudman. The state of authorship attribution studies: Some problems and solutions. *Computers and the Humanities*, 31:351–365, 1998.

[55] Hataichanok Saevanee, Nathan Clarke, Steven Furnell, and Valerio Biscione. Text-based active authentication for mobile devices. In *ICT Systems Security and Privacy Protection*, pages 99–112. Springer, 2014.

[56] Chao Shen, Zhongmin Cai, Xiaohong Guan, and Jialin Wang. On the effectiveness and applicability of mouse dynamics biometric for static authentication: A benchmark study. In *Biometrics (ICB), 2012 5th IAPR International Conference on*, pages 378–383. IEEE, 2012.

[57] T. Sim, S. Zhang, R. Janakiraman, and S. Kumar. Continuous verification using multimodal biometrics. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(4):687–700, 2007.

[58] Daniele Soria, Jonathan M Garibaldi, Federico Ambrogi, Elia M Biganzoli, and Ian O Ellis. A non-parametricversion of the naive bayes classifier. *Knowledge-Based Systems*, 24(6):775–784, 2011.

[59] Efstathios Stamatatos. A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–56, 2009.

[60] Esfathios Stamatatos. Title not available at press time. *Brooklyn Law School Journal of Law and Policy*, Forthcoming.

[61] Bo Sun, Fei Yu, Kui Wu, and Victor Leung. Mobility-based anomaly detection in cellular mobile networks. In *Proceedings of the 3rd ACM workshop on Wireless security*, pages 61–69. ACM, 2004.

[62] Robert R. Tenney and JR. Nils R. Sandell. Decision with distributed sensors. *IEEE Transactions on Aerospace and Electronic Systems*, AES-17:501–510, 1981.

[63] D. Umphress and G. Williams. Identity verification through keyboard characteristics. *International journal of man-machine studies*, 23(3):263–273, 1985.

[64] Dirk Van Bruggen, Shu Liu, Mitch Kajzer, Aaron Striegel, Charles R Crowell, and John D'Arcy. Modifying smartphone user locking behavior. In *Proceedings of the Ninth Symposium on Usable Privacy and Security*, page 10. ACM, 2013.

[65] Hans van Halteren. Author verification by linguistic profiling: An exploration of the parameter space. *ACM Transactions on Speech and Language Processing*, 4, 2007.

[66] Hans van Halteren, R. Harald Baayen, Fiona Tweedie, Marco Haverkort, and Anneke Neijt. New machine learning methods demonstrate the existence of a human stylome. *Journal of Quantitative Linguistics*, 12(1):65–77, 2005.

[67] R.V. Yampolskiy. Behavioral modeling: an overview. *American Journal of Applied Sciences*, 5(5):496–503, 2008.

[68] Nan Zheng, Aaron Paloski, and Haining Wang. An efficient user verification system via mouse movements. In *Proceedings of the 18th ACM conference on Computer and communications security*, CCS '11, pages 139–150, New York, NY, USA, 2011. ACM.

[69] R. Zheng, J. Li, H. Chen, , and Z. Huang. A framework for authorship identification of online messages: Writing-style features and classification techniques. *Journal of the American Society for Information Science and Technology*, 57(3):378–393, 2006.