AFRL-RI-RS-TR-2013-248

ACTIVE AUTHENTICATION LINGUISTIC MODALITIES

DREXEL UNIVERSITY

DECEMBER 2013

FINAL TECHNICAL REPORT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

STINFO COPY

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE

■ **AIR FORCE MATERIEL COMMAND** ■ **UNITED STATES AIR FORCE** ■ **ROME, NY 13441**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the 88th ABW, Wright-Patterson AFB Public Affairs Office and is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RI-RS-TR-2013-248 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

/ S /

JAMES M. NAGY
Work Unit Manager

/ S /

MICHAEL J. WESSING, Deputy Chief
Information Intelligence Systems &
Analysis Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
DECEMBER 2013	FINAL TECHNICAL REPORT	JUN 2012 – OCT 2013

4. TITLE AND SUBTITLE

ACTIVE AUTHENTICATION LINGUISTIC MODALITIES

5a. CONTRACT NUMBER
FA8750-12-C-0212

5b. GRANT NUMBER
N/A

5c. PROGRAM ELEMENT NUMBER
62304E

6. AUTHOR(S)
Alex Fridman, Ariel Stolerman, Sayandeep Acharya, Patrick Brennan, Patrick Juola, Rachel Greenstadt, Moshe Kam

5d. PROJECT NUMBER
ATAU

5e. TASK NUMBER
DR

5f. WORK UNIT NUMBER
XU

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Drexel University
3141 Chestnut Street
Philadelphia PA 19104

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory/RIEA
525 Brooks Road
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RI

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-RI-RS-TR-2013-248

12. DISTRIBUTION AVAILABILITY STATEMENT
Approved for Public Release; Distribution Unlimited. PA# 88ABW-2013-5387
Date Cleared: 18 DEC 2013

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Active authentication is the process of continuously verifying a user based on their on-going interaction with the computer. In this report, we consider a representative collection of behavioral biometrics: low-level modalities of keystroke dynamics and mouse movement, high-level modalities of stylometry and web browsing behavior. We develop a sensor for each modality and organize the sensors as a parallel binary detection decision fusion architecture. The decisions of each sensor (legitimate/illegitimate user) are fed into a Decision Fusion Center (DFC) which applies the Chair-Varshney fusion algorithm to generate a global decision. The DFC minimizes the probability of error using the local-sensor False Rejection Rates (FAR) and False Acceptance Rates (FRR) as well as the a-priori probability that user is legitimate to form the decision rule. We test our approach on a dataset collected from 67 users, each working individually in an office environment for a period of one week. We show that the fusion algorithm achieves lower probability of error than that of the best individual sensor in the fused set, and we are able to quantify the contribution of each modality to the overall performance. We consider the temporal characteristics of intruder detection, showing results of the time it takes to detect a change in user. We measure the effect of perfect adversarial compromise of sensors as part of the fusion system. Lastly, we consider a higher level classification model of users based on their personality metrics.

15. SUBJECT TERMS
Biometric modalities, software biometrics used for identification, computational linguistics, structural semanic analysis, forensic authorship, keystroke patterns, mouse movement used to validate users

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			**JAMES M. NAGY**
U	U	U	UU	48	19b. TELEPONE NUMBER (*Include area code*) **N/A**

TABLE OF **CONTENTS**

LIST OF FIGURES

LIST OF TABLES

1.0 SUMMARY

Active authentication is the process of continuously verifying a user based on their on-going interaction with the computer. In this report, we consider a representative collection of behavioral biometrics: low-level modalities of keystroke dynamics and mouse movement, high-level modalities of stylometry and web browsing behavior. We develop a sensor for each modality and organize the sensors as a parallel binary detection decision fusion architecture. The decisions of each sensor (legitimate/illegitimate user) are fed into a Decision Fusion Center (DFC) which applies the Chair-Varshney fusion algorithm to generate a global decision. The DFC minimizes the probability of error using the local-sensor False Rejection Rates (FRR) and False Acceptance Rates (FAR) as well as the a-priori probability that user is legitimate to form the decision rule. We test our approach on a dataset collected from 67 users, each working individually in an office environment for a period of one week. We show that the fusion algorithm achieves lower probability of error than that of the best individual sensor in the fused set, and we are able to quantify the contribution of each modality to the overall performance. We consider the temporal characteristics of intruder detection, showing results of the time it takes to detect a change in user. We measure the effect of perfect adversarial compromise of sensors as part of the fusion system. Lastly, we consider a higher level classification model of users based on their personality metrics.

2.0 INTRODUCTION

The challenge of identity verification for the purpose of access control 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 [1]. These systems rely on input 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 [2], showing error rates ranging from 0% [3] to 30% [4] 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. In particular, to represent this general real-world scenario, we created a simulated office environment in order to collect behavioral biometrics associated with typical human-computer interaction (HCI) by an office worker.

In this report, 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 selection. In other words, we seek to motivate the community to search not for the perfect biometric sensor, but for a large collection of good ones. 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 [5].

We employ four classes of biometrics: keystroke dynamics, mouse movement, web browsing behavior, and stylometry. The latter two have not been considered in literature, to the best of our knowledge, in the continuous authentication context. Stylometric analysis, in particular, is well established (and accurate enough to be admissible as legal evidence [6], [7]) but its application to continuous verification of user identity is novel. Based on the success of authorship attribution in other fields, we seek to characterize its performance in this much more dynamic and time-constrained problem space.

A key issue to explore in this project is what linguistic level or modality is most informative and robust. It is relatively easy, for example, for a person to deliberately use specific words (such as UK or US spellings), but harder to control the use of specific function words or the exact frequency of character digraphs. We therefore specifically propose a multimodal analysis at several levels ranging from the purely mechanical (character n-grams) to higher order linguistic constructions such as concepts or structure. Indeed, the best results may (and probably will) come from a combination of several types of data, carefully fused. Our proposed framework makes this type of multimodal analysis both practical and effective.

2.1 Decision Fusion

We further propose to use decision fusion in order to test and validate the linguistics-based authentication scheme and provide accurate assessment of the circumstances and conditions under which it can contribute to user authentication. Due to the constraint of using only existing DoD standard computing gear at this stage of the research, the task of active authentication in this effort is limited to (1) tracking and developing inference from previously-proposed characteristics (or metrics) that fit the computing gear restrictions; (2) using new modalities, such as linguistics-based authentication and integrate them in a suite that includes previously proposed characteristics; and (3) augment the suite by developing inference from the loyalty of the user to software and human-machine interface technologies from the menu of technologies available to him/her. Under the first category we include characteristics such as usage of websites and other on-line resources and the related metrics. These characterize viewing habits [8]; behavior pattern within social networks [9] and search habits in databases and electronic libraries [10]. Of potential importance here are characterizations of groups of users in a way that allows the flagging out of individuals or subgroups that deviate from common patterns [11]. Also of importance are mouse movements [12]; and typing and keystroke patterns [13]. Under the second category we focus on forensic linguistics based criteria; under the third category we include semi permanent habits such as user attachment to certain web browsers, search engines and financial/management software; frequency of visits to most popular websites; and computer file naming habits [14], [15]. Our objective is to put (1) - (3) in a common frame of reference (a decision fusion system) and use this system through extensive testing to assess the new modalities developed in this project for accuracy, user detectability, reaction speed, convergence rate, data requirements, stability and consistency. Moreover we would like to measure and document resistance of the different modalities, new and old, to *deception*.

2.2 Fusion of Biometric Classifiers

A defining problem of active authentication arises from the fact that a verification of identity must be carried out continuously on a sample of sensor 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 sensors may provide more data than others. For example, as the user browses the web, the mouse and web browsing sensors will be actively flooded with data, while the keystroke dynamics and stylometry sensors may only get a few infrequent key presses. This motivates the recent work on multimodal authentication systems where the decisions of multiple classifiers are fused together [16]. 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 [17]. When neural networks are used as classifiers, an ensemble of sensors is constructed and fused based on different initialization of the neural network [18]. Our approach in this report is to apply the Chair-Varshney optimal fusion rule [19] for the combination of available multimodal decisions. Furthermore, we are motivated by the work in [20] that greater reduction in error rates is achieved when the classifiers are distinctly different (i.e. using different behavioral biometrics).

In this study we model the authentication problem as a binary hypothesis testing. The null hypothesis H0 is that the user is illegitimate; the alternative hypothesis H1 is that the user is legitimate. The tradeoff between the resulting False Rejection Rate (P^M) and the False

Acceptance Rate (P^A) can be mediated by tuning the weights in a Bayesian cost function or adopting a Neymann Pearson detection philosophy whereby the False Rejection Rate (the rate at which the legitimate user is rejected as illegitimate) is capped at a certain bound (representing a judgment on how distracting that system can be to the legitimate user). The False Acceptance Rate (the rate at which illegitimate users will be recognized as legitimate) is then minimized under the False Rejection Rate bound constraint.

Figure 1 shows the process of going from raw keyboard and mouse data to an authenticating decision, using the HCI suite and two stylometry suites as examples. It process a stream of asynchronous events from the keyboard and mouse, extracts the features needed by the HCI suite and the stylometry suites. The classification based on the individual set of features is fused to produce an authenticating decision.

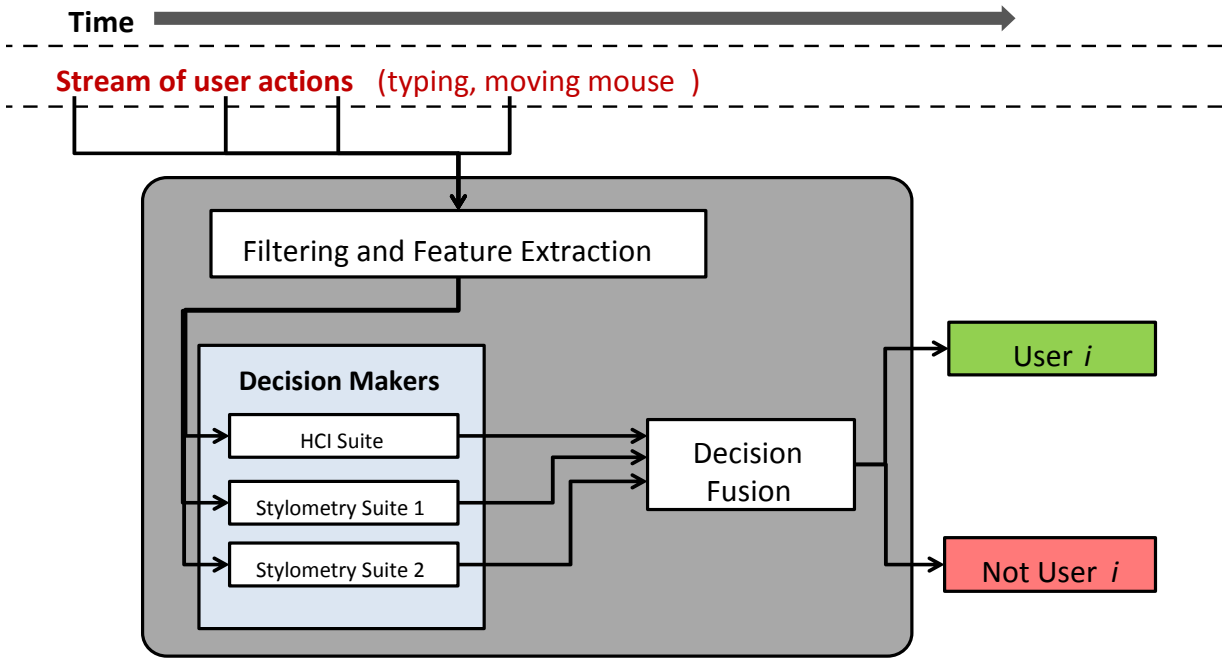


Figure 1: Temporal view of the authentication system. Module for User *i*

2.3 Challenges and Limitations

An active authentication system presents a few concerns. First, a potential performance overhead is expected to accompany deployment of such systems, as they require constant monitoring and logging of user input, and on-the-fly processing of all the sensor components of the system. Since multiple sensors are used, and some of them may require large amounts of memory and computing power, a careful configuration should be applied to balance the tradeoff between the accuracy of the system and its expected resource consumption behavior.

Another concern with this type of authentication system is its user input requirements. In non-active authentication schemes, the user is required to provide credentials only when logging in, and perhaps when certain operations are to be executed. The provided credentials consist of some sort of personal key (password, private key etc.), dedicated for the purpose of identifying the system's users. In active authentication systems based on modalities presented in this report, all of the user computer interaction input is required: mouse movements, keyboard usage and web browsing behavior. The precise sequence and timing of mouse and keyboard events is essential for the system's performance. However, this type of input is not designed for authentication, and in most probability contains sensitive and private information, collected when the user types in passphrases to log into accounts, writes something personal s/he wishes to keep confidential, or simply browses the web. To cope with these security and privacy issues, some actions can be taken in the design of such a system: the collected data should be managed carefully, by avoiding storage of raw collected data (i.e. save only parsed feature vectors extracted from the data); use encrypted storage for the data that is stored.

If privacy is of primary importance during specific period of time, a fusion system may activate only a subset of the detectors. For example, one of the benefits of mouse movement as a biometric for authentication is, unlike keystroke dynamics, it does not capture any sensitive information. The mouse only provides us information about *how* the user was using the device, and cannot be processed to reconstruct *what* the user was accomplishing with it on the screen. Therefore, in a privacy-constrained system, the fusion center may utilize only mouse-based metrics to verify the user.

3.0 METHODS, ASSUMPTIONS AND PROCEDURES

3.1 Simulated Work Environment Dataset

The source of behavioral biometrics data we utilized for the multi-modal fusion and identity verification in this report 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 four 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 extensively 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.

There were three data files produced by two tracking applications. In sum, they contain the following data:

- Mouse movement, mouse click, and mouse scroll wheel events at a granularity of 5 miliseconds.
- Keystroke dynamics (include press, hold, release durations) for all keyboard keys including special keys at a granularity of 5 miliseconds.
- 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.
- Web browser url and page title at a granularity of 1 second. The title of the page often contains a rich information about the status of the user's interaction with the website. For example, for web mail sites, the title contains the number of unread emails.

Table 1: Statisitics on biometric data

Metric	Total	Per User-Day
Websites Visited	40,224	437
Mouse move events	9,819,421	106,733
Mouse clicks	178,349	1,938
Scroll wheel events	404,531	4,397
Keystroke events	1,243,286	13,514

Table 1 shows statistics on the biometric data in the corpus. The table is Statistics on the 19-user subset of the biometric data contained in the dataset. The data is aggregated over 92 days of data. The table contains data aggregated over all 80 users. It also shows the average amount of data available per user per day. 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 1, we count just the down press and not the release. The general conclusions drawn from observing these statistics is that the users were very active in using their mouse in browsing the web, averaging 55 web sites visited per hour and 242 mouse clicks per hour.

As an example of the variation in the dataset, Figure 2 shows a heat map visualization of the aggregate first day mouse movements for 14 of the 19 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 to create unique behavioral profiles. The behavioral profiles show 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.

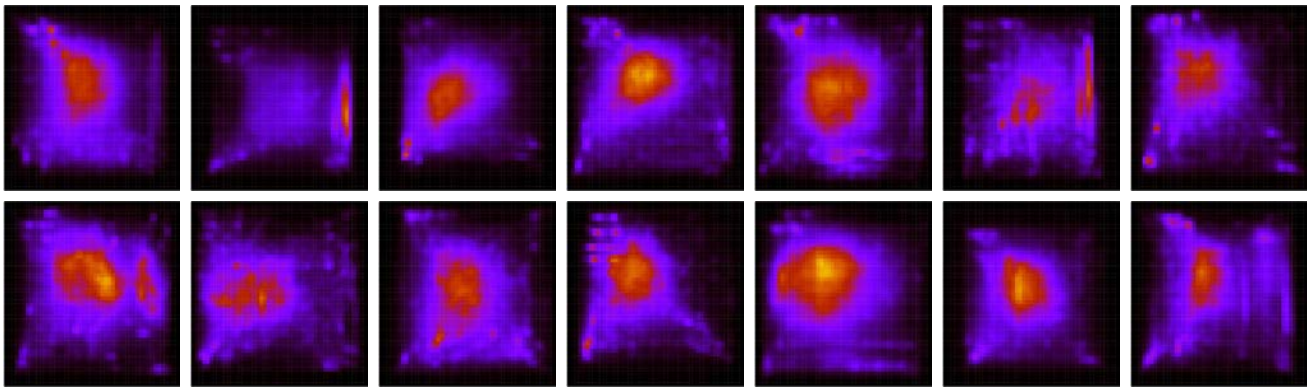


Figure 2: Map visualization of aggregate mouse movements

3.2 Biometric Sensors

The sensors we consider in this report span across different levels and directions for profiling: linguistic style (stylometry), mouse movement patterns, keystroke dynamics and web browsing behavior. Each sensor type works differently in terms of required amount of input data, type of collected data (mouse events, keystrokes, and different usage statistics) and performance. Sections 3.2.1 to 3.2.3 present each of these four modalities, discuss their configurations and provide standalone evaluations on the collected dataset.

We broadly categorize these sensors according to the degree of conscious cognitive involvement measured by the sensors. The distinction can be thought of as that between "how" and "what". We refer to the mouse movement and keystroke dynamics sensors 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 sensors are "high-level" because they track *what* we click on with the mouse and *what* we type. The following are the categories and abbreviations of the sensors presented, evaluated, and fused in this report:

- Low-level sensors:
  - M1: mouse curvature angle
  - M2: mouse curvature distance
  - M3: mouse direction
  - K1: keystroke interval time
  - K2: keystroke dwell time
- High-level sensors:
  - W1: website domain visit frequency
  - S1: stylometry (1000 char., 30 min. window)
  - S2: stylometry (500 char., 30 min. window)
  - S3: stylometry (400 char., 10 min. window)
  - S4: stylometry (100 char., 10 min. window)

Although each modality is configured differently, some common configurations and evaluations are applied with all the sensors to be used in data fusion (Section 3.3):

- *Dataset parsing*: The data is divided into non-overlapping windows, of the following lengths: 5, 10, 15, 20, 25, 30 and 60 minutes. The sliding windows technique simulates actual data input behavior in a real-time authentication system.
- *Evaluation*: The common evaluation method used with each sensor for data fusion is measuring the averaged error rates across five experiments; In each experiment, data of 4 days is taken for training and the remaining day – for testing. The False Acceptance Rate (FAR) and False Rejection Rate (FRR) are taken as input for the fusion system, as a measurement of the expected performance of the sensors. Each experiment consists of three phases:
  1. Train the classifier(s) using the training set
  2. Determine FAR and FRR based on the training set
  3. Classify the windows in the test set.

Phases 1 and 2 of evaluation mentioned above differ between the stylometry sensors and the others: for the stylometry sensors, all 4 days are used for training, and the FAR/FRR are set by the results of 10-folds cross validation on the training set itself; for all other sensors, 3 of the 4 training days are taken for actual training and the FAR/FRR are set by testing the results of

classifying the fourth day. The test phase (3) is the same for all sensors: classify the windows of the fifth day.

3.2.1 Stylometry.

3.2.1.1 Background.

Authorship attribution based on linguistic style, or Stylometry, is a well-researched field [21], [22], [23], [24], [25], [26]. 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" [27]) 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 [28], [29], grammar [30], character n-grams [31] 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 [32], [33], 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 [34], [21], [35], [36], [37] 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 rates. The unmasking technique [38] has been shown effective on a dataset of 21 books of 10 different 19th-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

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

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.

3.2.1.2 Configuration.

For this report we chose the simplest setting of closed-world stylometry: we use classifiers trained on the closed set of users, where each classification results with one of those users as the author.

In the preprocessing phase, we parsed the keystrokes log files to produce a list of documents consisting of non-overlapping windows for each user, with time-based sizes spanning from 5-minutes to 1-hour windows, as mentioned above. Specifically for stylometry-based biometric, selecting the size of the window affects a delicate tradeoff between the amount of captured text (and probability for correct style-profiling of that window) and response time of the system, whereas other biometrics detailed in the following sections can perform satisfactorily with small windows (even the size of seconds). During preprocessing, only keystrokes were taken (mouse events and key releases were filtered out) 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 chosen feature set is probably the most crucial part of the configuration. The constructed feature set, denoted the *AA* feature set hereinafter, is a variation of the *Writeprints* [39] 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 2. By using a rich linguistic feature set we are able to better 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 (or lack of decisiveness).

The features were extracted using the *JStylo* framework [2] [40], an open-source authorship attribution platform developed in the Privacy, Security and Automation Laboratory at Drexel University. JStylo was chosen for analysis 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

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

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 popular data mining and machine learning platform Weka [41], which we utilized for the classification process.

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 preprocessing tool developed for this unique dataset, that applies the relevant special characters on the text. For instance, the character sequence ch*ββ*Cch*ββ*hicago becomes Chicago, where *β* represents backspace.

Table 2: The *AA* feature set.

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

Second, since the windows are determined by time and not amount of collected data, normalization is crucial for all frequency-based features (which consist of the majority of the features). These features were simply divided by the most relevant measurement related to the feature. For instance, character bigrams were divided by the total character count of the window.

For classification we used sequential minimal optimization (SMO) support vector machines [42] with polynomial kernel, available in Weka. Support vector machines are commonly used for authorship attribution [43], [44], [45] and known to achieve high performance and accuracy.

Finally, the data was analyzed with the stylometry sensor 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 sensor output no decision. The different thresholds allow us to assess the tradeoff in the sensor'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 sensor less available. Note that even the largest threshold (1000) is considerably smaller than used in most previous stylometry analyses – a minimum of 500 *words*. Along with the varying time-wise window size, a matrix of configurations is set for this sensor, out of which a few were chosen for the fusion system, as detailed in section 3.3.

3.2.2 Low-Level Metrics.

3.2.2.1 Background.

Keystroke dynamics is one of the most extensively studied topics in behavioral biometrics [46]. The feature space that has been investigated ranges from the simple metrics of key press interval [47] and dwell [48] times to multi-key features such as trigraph duration with an allowance for typing errors [2]. Furthermore, a large amount of classification methods have been studied for mapping these features into authentication decisions. Broadly these approachs fall in one of two categories: statistical methods [49] and neural networks [50], 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 [51] or required an excessively large static corpus of mouse movement data to achieve good results [1], where an FAR and FRR of 0.0246 is achieved from a testing window of 2000 mouse actions. The work in [32] 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 sensor 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.

3.2.2.2 Configuration.

The low-level metrics of keystroke and mouse dynamics detectors, along with the domain visit frequency detector, all use support vector machines (SVMs) but a different implementation than used by the stylometry detectors in section 3.2.1.2. For the training and testing of each individual binary classification detector we utilize an OpenCV C++ interface to LIBSVM [52], [53], [54] using the radial basis function as the kernel.

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 report, we consider three metrics based on those described in [32]: (M1) curvature angle, (M2) curvature distance, and (M3) movement direction. The last is computed from a single tuplet, and the former two are computed from two adjacent tuplets.

The mouse movement curvature metrics in [32] end in a mouse click by definition. We consider a much higher density of mouse movement events, including those that do not end in a button click, but at the cost that some of these movement events may not represent any real intent from the user and thus essentially provide noise to the sensor.

We chose two of the simplest and most frequently occurring keystroke dynamics features: (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 is a strictly positive number, the interval K1 can be negative if another key is pressed before a prior one is released.

3.2.3 Web Browsing Behavior.

Web browsing behavior has been studied extensively in literature [55] but not in the context of active authentication. We used the same classification as for low-level sensors described in section 3.2.2.2, and the feature vector of the visit frequency to the 20 most visited websites in the dataset (as shown in Table 3). The frequency of visits to each of these domains is used as the feature vector based on which a user's web browsing profile is built.

Table 3: Top twenty websites visited by the users in the dataset

www.google.com	7.0%
www.bing.com	7.0%
www.facebook.com	5.0%
search.yahoo.com	4.1%
en.wikipedia.org	2.9%
dell.msn.com	2.4%
www.youtube.com	2.4%
www.pandora.com	2.2%
www.yahoo.com	1.3%
sites.google.com	1.1%
disneyworld.disney.go.com	1.0%
pinterest.com	0.9%
pittsburgh.about.com	0.9%
www.last.fm	0.9%
duquesne.mrooms2.net	0.9%
us.mg5.mail.yahoo.com	0.8%
tvtropes.org	0.7%
www.urbanspoon.com	0.7%
www2.timesdispatch.com	0.7%
mail.google.com	0.6%

Figure 3 shows the FAR and FRR rates for each of the 19 users for a 10 minute window. A classifier did not generate a decision when less than 10 website were visited in that window. This sensor achieves reasonably low error rates, but has shown to degrade in performance as the user base grows. With a larger number of users, the top twenty websites tend to become more generic and thus are not good features based on which to verify a user's identity.

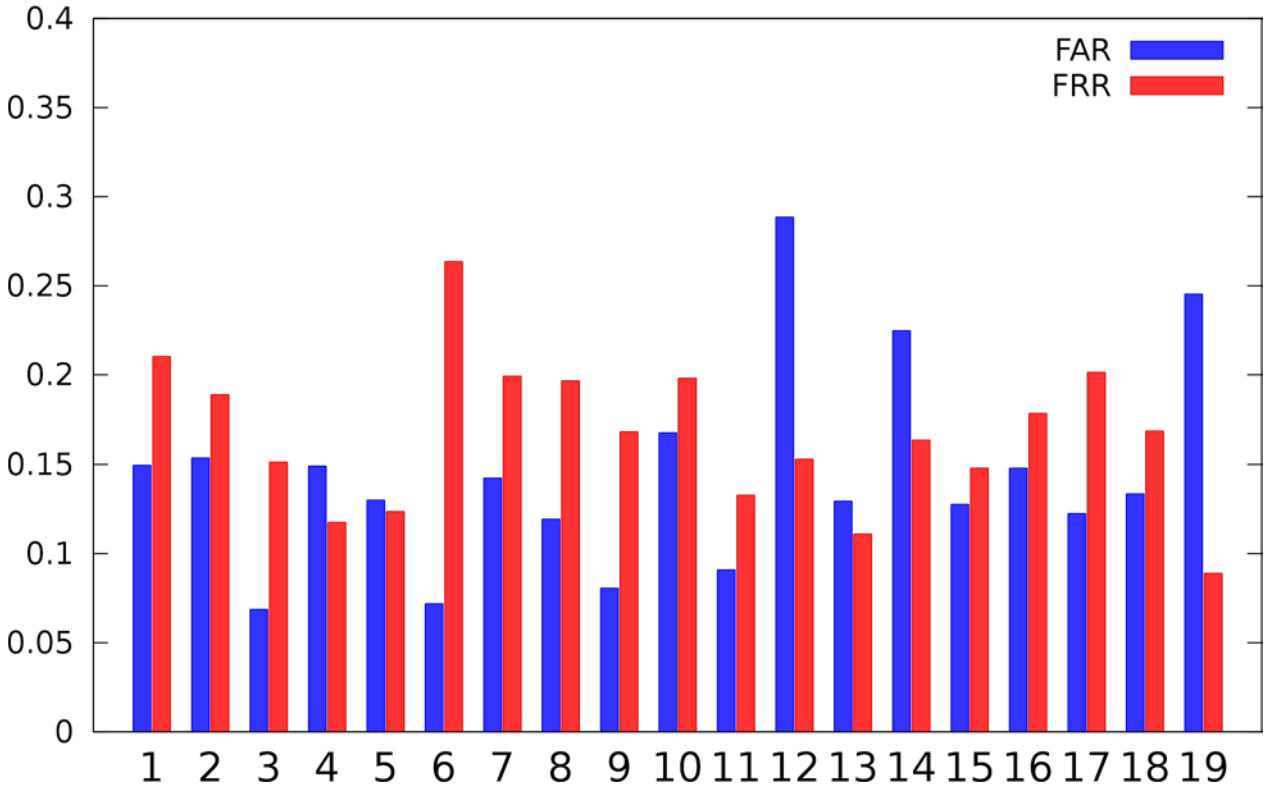


Figure 3: FAR and FRR rates using the web domain frequency sensor

3.2.4 Classification Based on Personality Characteristics.

3.2.4.1 Background.

The same technologies [56], [57], [23], [24], [25] that make specific identification of authorship possible can also be used to infer group characteristics of the author, such as demographics [58], personality [59], and first-language [60]. One attribute of our research is to apply these classifications to a security context. Put simply, if the authorized user is a 40 year old extroverted English-speaking female, but the person at the keyboard is a 21 year old introverted Russian-speaking male, there may be a problem to be investigated.

One advantage of this approach is that it may be less equipment-dependent than low-level metrics such as keyboard dynamics; changing the keyboard will not change the age or gender of the person sitting at the keyboard. Similarly, this approach may help investigators follow up on a security incident (for example, the analysis above could provide a starting point for law enforcement if they want to find the intruder). At the same time, this kind of analysis may require substantially more data for a reliable classification. While this would be a potentially crippling weakness in a standalone security product, it is not a serious drawback in a suitable information fusion framework.

3.2.4.2 Configuration.

As part of the simulated work environment described above, participants were asked (on the morning of the first day) to take a variety of standard personality tests, including a basic demographic survey (incorporating *inter alia* gender, education level, native language, age, and dominant hand), the Rosenberg Self-Esteem Scale, the Myers-Briggs Personality Inventory (MBTI), the NEO PI-R, the Multiple Intelligences Developmental Assessment Scales (MIDAS), and the Learning Styles Inventory. These provided nominal and/or numeric data on each participant as ground truths along a variety of dimensions. As with the stylometric analysis performed above, we used low-level character- or wordbased n-grams to develop a profile of a "typical" user of each type. Using leave-one-out cross-validation, we tested varying sized chunks of data for each volunteer participant against the other participants in the study.

The specific instruments used are as follows:

- **Learning Styles**: The Learning Styles Inventory Test was used to determine how well a person learns from a list of 7 methods, such as Verbal, Visual, and so forth see below for details).
- **Gender**: For this experiment, gender consisted of two categories, male and female, as reported by self-identification.
- **Self-Esteem**: The Rosenberg Self-Esteem Scale provides a numerical measure reflecting the testtaker's self-esteem; we categorized this score into four categories (Very Low, Low, High, Very High) and assigned participants to the corresponding categories (thus producing a nominal task from an ordinal one).
- **Myers-Briggs**: The Myers-Briggs Type Inventory (MBTI) categorizes a person into four binary categories (listed below) to reflect their personality. In this set of experiments we attempted to identify which half of each category the volunteer participant would be in.
- **MIDAS**: The Multiple Intelligences Developmental Assessment Scales provides a numerical score to reflect the test-taker's level of intelligence in a number of different categories and subcategories (listed below). In this set of experiments we attempted to identify the category and subcategory on which the participant achieved the highest score (i.e., determine the participant's specific strengths).

3.2.4.3 Evaluation

The analytic results are presented in table 4 and table 5. The best performing results were obtained using the combination of preprocessors, features, analysis methods, and distance functions listed, with both results and baseline performance (obtained by always choosing the most frequent outcome) presented.

Table 4: Results of best-performing high level analyses

Modality	Base Accuracy	Best Performing Method Accuracy
Learning Styles Inventory	30.40%	78.77%
MBTI - E/I	56.94%	82.02%
MBTI - S/N	52.11%	80.92%
MBTI - T/F	59.15%	79.62%
MBTI - J/P	50.70%	83.57%
Rosenberg Self-Esteem	57.50%	80.47%
MIDAS - Primary Catagories	22.10%	70.74%
MIDAS - Interpersonal	48%	80.80%
MIDAS - Intrapersonal	40%	82%
MIDAS - Kinesthetic	53.24%	88.60%
MIDAS - Leadership	45.07%	84.40%
MIDAS - Linguistic	69.33%	78.90%
MIDAS - Logical/Mathematical	38.89%	79.50%
MIDAS - Musical	40.85%	75.20%
MIDAS - Naturalist	38.46%	81.60%
MIDAS - Spatial	45%	80.00%
Gender	66.25%	86.91%

Table 5: Best-performing methods for high-level attribution

Modality	Canonicizers	Features	Analysis Method	Distance Function
Learning Styles Inventory	"Normalize Whitespace, Unify Case"	Character 16grams	Centroid-Based Nearest Neighbor	Alt Intersection
MBTI - E/I	"Normalize Whitespace, Unify Case"	Character 12grams	Centroid-Based Nearest Neighbor	Alt Intersection
MBTI - S/N	"Normalize Whitespace, Unify Case"	Character 13grams	Centroid-Based Nearest Neighbor	Alt Intersection
MBTI - T/F	"Normalize Whitespace, Unify Case"	Character 14grams	Centroid-Based Nearest Neighbor	Alt Intersection
MBTI - J/P	"Normalize Whitespace, Unify Case"	Character 11grams	Centroid-Based Nearest Neighbor	Alt Intersection
Rosenberg Self-Esteem	"Normalize Whitespace, Unify Case"	Character 15grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS - Primary Catagories	"Normalize Whitespace, Unify Case"	Character 15grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS - Interpersonal	"Normalize Whitespace, Unify Case"	Character 11grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS- Intrapersonal	"Normalize Whitespace, Unify Case"	Character 9grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS - Kinesthetic	"Normalize Whitespace, Unify Case"	Character 8grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS- Leadership	"Normalize Whitespace, Unify Case"	Character 11grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS - Linguistic	"Normalize Whitespace, Unify Case"	Character 4grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS - Logical/ Mathematical	"Normalize Whitespace, Unify Case"	Character 11grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS - Musical	"Normalize Whitespace, Unify Case"	Character 11grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS - Naturalist	"Normalize Whitespace, Unify Case"	Character 11grams	Centroid-Based Nearest Neighbor	Alt Intersection
MIDAS - Spatial	"Normalize Whitespace, Unify Case"	Character 11grams	Centroid-Based Nearest Neighbor	Alt Intersection
Gender	"Normalize Whitespace, Unify Case"	Character 10grams	Centroid-Based Nearest Neighbor	Alt Intersection

3.4 Decision Fusion

The motivation for the use of multiple sensors to detect an event is to harness the power of the sensors to provide an accurate assessment of the environment, which a single sensor may not be able to provide. In centralized architectures, raw data from all sensors 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 its more convenient to fuse their decisions rather than re-create them at the fusion center. In the distributed architectures, some processing of data is performed at each sensor, and the resulting information is sent out from each sensor 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 sensors is described by Tenney and Sandell in [61] who studied a parallel decision architecture. As described in [62], 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($u = 1$). Figure 4 shows the architecture and the associated symbols. Tenney and Sandell [61] and Reibman and Nolte [63] 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 (LDMs) and the DFC are co-dependent. In most scenarios the resulting complexity renders the quest for an optimal design impractical.

Chair and Varshney in [19] 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 [62], 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

$$B^{(k)}(C_{00},C_{01},C_{10},C_{11}) = C^{(k)}_{00}Pr(H_0,D_0) + C^{(k)}_{10}Pr(H_0;D_1) + C^{(k)}_{01}Pr(H_1;D_0) + C^{(k)}11Pr(H_1;D_1) \quad (1)$$

where $C_{00}^{(k)}, C_{01}^{(k)}, C_{10}^{(k)}, C_{11}^{(k)}$ are the pre-specified 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 report

$$C^{(k)}_{00} = C^{(k)}_{11} = 0,\ k = 1,2,...,n$$

$$C_{10}^{(k)} = C_{01}^{(k)} = 1,\ k = 1,2,...,n \qquad (2)$$

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

3.4.1 Fusion Rule

The parallel distributed fusion scheme (see Figure 4) allows each sensor 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 sensor sends out a decision of the form:

$$u_i = \begin{cases} 1, & \text{if } H_1 \text{ is decided} \\ -1, & \text{if } H_0 \text{ is decided} \end{cases} \qquad (3)$$

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

$$\frac{P(u_1,...,u_n|H_1)}{P(u_1,...,u_n|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0(C_{10}-C_{00})}{P_1(C_{01}-C_{11})} = \tau \qquad (4)$$

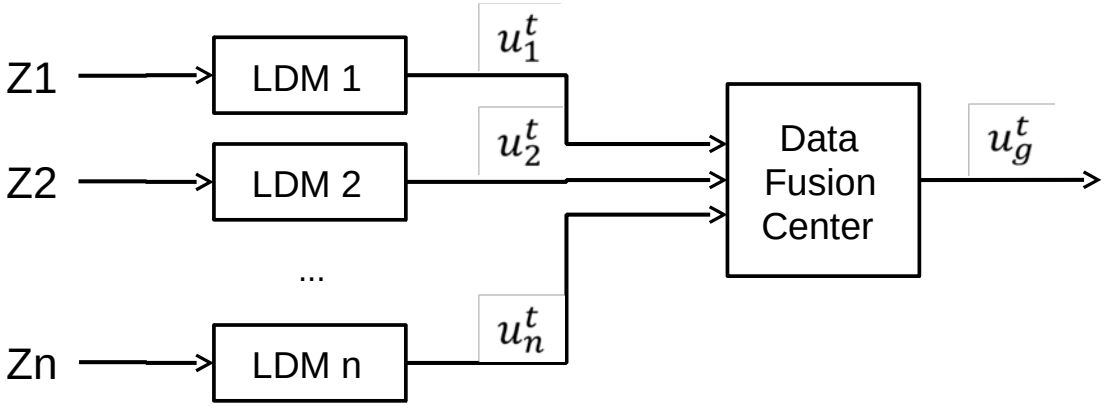


Figure 4: Architecture for the fusion of decentralized detectors

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 equation (2), the Bayes' risk

becomes total probability of error and the right hand side of equation (4) becomes $\frac{P_0}{P_1}$. In this case the general fusion rule proposed in [19] is

$$f(u_1,\ldots,u_n) = \begin{cases} 1, \text{ if } a_0 + \sum_{i=0}^{n} a_i u_i > 0 \\ -1, \text{ otherwise} \end{cases} \quad (5)$$

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

$$a_0 = \log \frac{P_1}{P_0} \quad \text{(6) and (7)}$$

$$a_i = \begin{cases} \log \frac{1-P_i^M}{P_i^F} \\ \log \frac{1-P_i^F}{P_i^M}, \end{cases}$$

Upper equation if $u_i = 1$ and lower equation if $u_i = -1$

Kam et al. in [62] developed expressions for the the global performance (global FAR and FRR) of the distributed system described above. The expressions for global error rates are given by:

3.4.1.1 Case 1: For non-identical sensors

(each having different FAR and FRR)

$$P_G^F = \sum_{h_1=0}^{1} \sum_{h_2=0}^{1} \cdots \sum_{h_n=0}^{1} \left| \prod_{i=1}^{n} (h_i - P_i^F) \right| \cdot U_{-1}\left[\prod_{i=1}^{n} \left(\frac{1-P_i^M}{P_i^F}\right)^{1-h_i} \cdot \left(\frac{P_i^M}{1-P_i^F}\right)^{h_i} - \tau \right] \quad (8)$$

$$P_G^M = \sum_{h_1=0}^{1} \sum_{h_2=0}^{1} \cdots \sum_{h_n=0}^{1} \left| \prod_{i=1}^{n} (h_i - P_i^M) \right| \cdot U_{-1}\left[\tau - \prod_{i=1}^{n} \left(\frac{1-P_i^M}{P_i^F}\right)^{h_i} \cdot \left(\frac{P_i^M}{1-P_i^F}\right)^{1-h_i} \right] \quad (9)$$

where U_{-1} is the unit step function such that

$$U_{-1}(x) = \begin{cases} 0, \text{ if } x < 0 \\ 1 \end{cases}$$

1 if $x \geq 0$

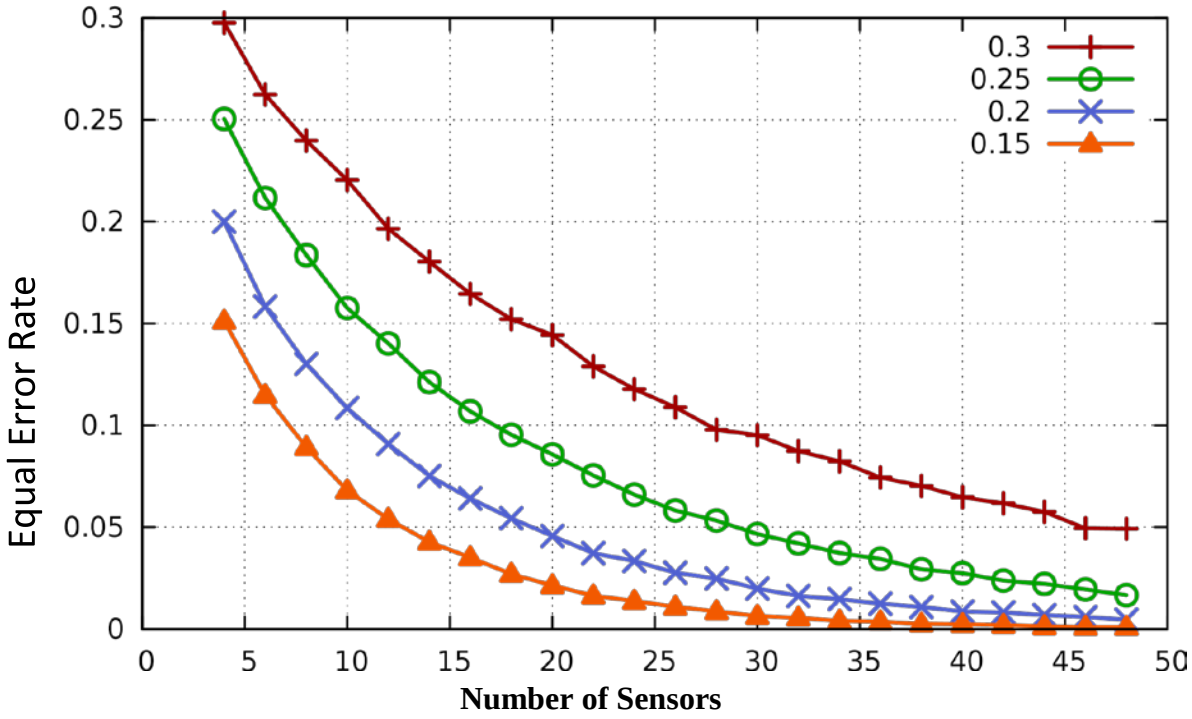


Figure 5: Application of Chair-Varshney fusion rule

The application of the Chair-Varshney fusion rule on *n* generated sensors ranging from equal error rates (EER) of 0.5 to 0.3, 0.25, 0.2, and 0.15, respectively. As the number of sensors increases, the fusion rule approaches an error rate of zero, outperforming the best sensor in the set for all data points.

3.4.1.2 Case 2: For identical sensors

(all sensors have same FAR and FRR)

Assuming all sensors error rates are same with $P_i^F = P^F$ and $P_i^M = P^M$, $i = 1,2,...,n$ the global error rates are

$$P_G^F = \sum_{i=J_F}^{n} \binom{n}{i} (P^F)^i (1-P^F)^{n-i} \qquad (10)$$

$$P_G^M = \sum_{i=J_M}^{n} \binom{n}{i} (P^M)^i (1-P^M)^{n-i} \qquad (11)$$

The bounds are given as

$$J_F = int\left[\frac{log(\tau)+n[log(1-P^F)-logP^M]}{[log(1-P^M)-logP^F]+[log(1-P^F)-logP^M]}\right] \qquad (12) \text{ and } (13)$$

$$J_M = int\left[\frac{n[log(1-P^M)-logP^F]-log(\tau)}{[log(1-P^M)-logP^F]+[log(1-P^F)-logP^M]}\right]$$

In the above expressions, *int*[*x*] represents the smallest integer larger than or equal to *x* and

$$\tau = \frac{P_0(C_{10}-C_{00})}{P_1(C_{01}-C_{11})}$$

where C_{ij} is the cost of deciding H_i when H_j is true. Note that when the objective function is the total probability of error, from (6) we have $a_0 = -log(\tau)$.

Figure 5 shows the Monte Carlo simulated probability of detections for randomly generated sensors as a function of the number of sensors (*n*), the theoretical values of which could be obtained from (8 - 11). For a group of *n* sensors used to compute each data point in Figure 5, the equal error rate (P_i^E) for each sensor $i \in \{1..n\}$ used to generate the decisions is set to:

$$P_i^E = \frac{i-1}{n-1}(P_{\max}^E - P_{\min}^E) + P_{\min}^E \qquad (14)$$

where $P_{\max}^E$ and $P_{\max}^E$ are the error rates of the worst and best performing sensors of the group, respectively. The purpose of providing a range of error rates in the group of generated sensors is so that it represents the similarly varying FAR and FRR rates of the biometric sensors considered in this report. The four sets of data in Figure 5 use $P_{\max}^E$ of 0.5 and a $P_{\min}^E$ of 0.3, 0.25, 0.2, 0.15. What we can observe from the figure is that for the fusion of multiple sensors has lower error rate than the best sensor in the group, and that the error rate decreases gradually as the number of sensors in the group increases.

3.4.2 Extendable Fusion Framework.

As is intuited in section 3.3.1, the performance of the fused global detector improves as the number of local sensors increases. Furthermore, it is shown in [20] that fusion of classifiers trained on distinct feature sets leads to greatest reduction in system error. In other words, an ideal active authentication system gathers input from as many different behavioral biometric sensors as possible. In designing the fusion system in this report, one of our privacy goals from the software engineering perspective was to provide an easy and clear way of adding sensors to the fusion system portfolio without having to know anything about how the system works.

In particular, the keystroke, mouse, and web browsing sensors are implemented in C++, while the stylometry sensor is implemented in Java. There are two ways to provide decision information to the fusion center. First is through C++ API, and the other is through a structured CSV file that contains a sequence of decisions produced by the sensor in the following format: timestamp, FAR, FRR, decision in $\{-1,0,1\}$. The decision is provided with respect to a particular user, where 1 indicates valid user, -1 indicates invalid user, and 0 indicating absence of a decision (usually due to insufficient data available in the time-window under consideration).

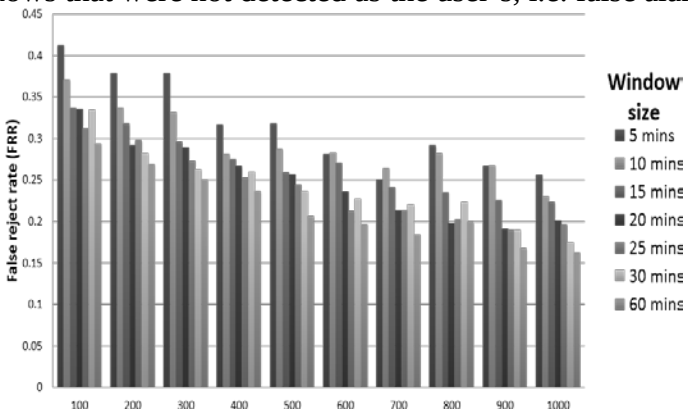
Using this file format, an arbitrary number of local decision makers can be added to the fusion system. It is assumed that they all provide their decisions asynchronously. The fusion algorithm decomposes all sensor output into a stream of decisions with associated FAR and FRR rates and fuses them, assigning less weight to the older information.

4.0 RESULTS AND DISCUSSION

4.1 Stylometry

To evaluate the performance of the stylometry sensor as a standlone sensor, we use 10-folds cross validation on the entire dataset. False reject rate (FRR) and false accept rate (FAR) results are shown in Figure 6 and Figure 7, respectively. Figure 8 illustrates the percentage of remaining windows, after removing all those that do not pass the minimum characters-per-window threshold.

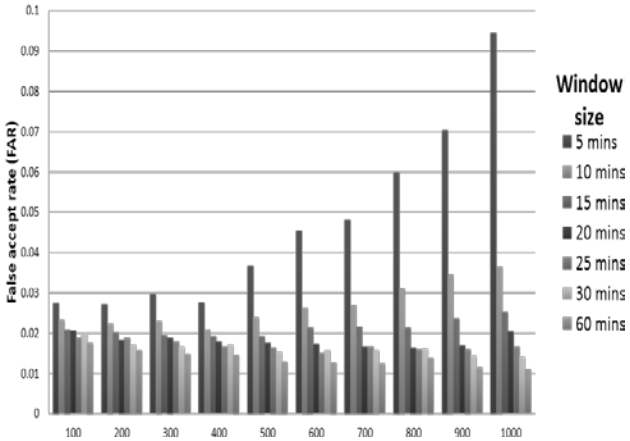
Figure 6 shows Weighted avg. false reject rate (FRR) for 10-folds cross-validation using the stylometry sensor with varying time-wise window sizes and varying threshold for minimum number of characters per window. Only windows that pass the threshold (i.e. contain at least that many characters) participated in the analysis. This measurement accounts for the portion of legitimate user's windows that were not detected as the user's, i.e. false alarms.



Min. characters per window

Figure 6: Weighted avg. false reject rate (FRR)

Figure 7 shows Weighted avg. false accept rate (FAR) for 10-folds cross-validation using the stylometry sensor, with the same configurations as described in Figure 6. The FAR accounts for the portion of intruder windows that were classified as the legitimate user's, i.e. security breaches.



Min. characters per window

Figure 7: Weighted avg. false accept rate (FAR)

Figure 8 shows Percentage of remaining windows out of the total windows after filtering by the minimum characters-per-window threshold.

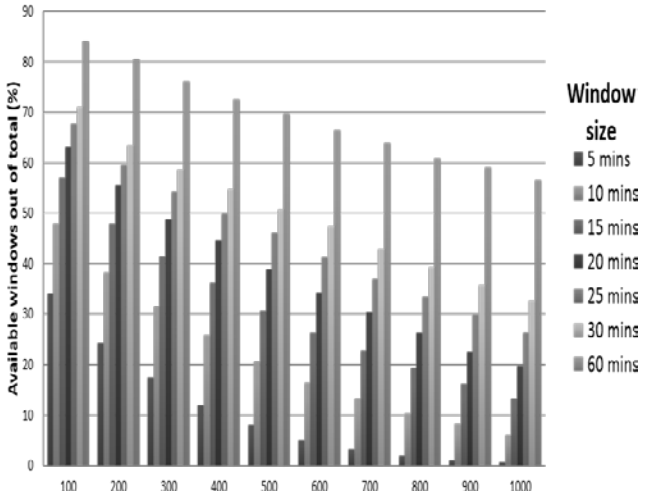
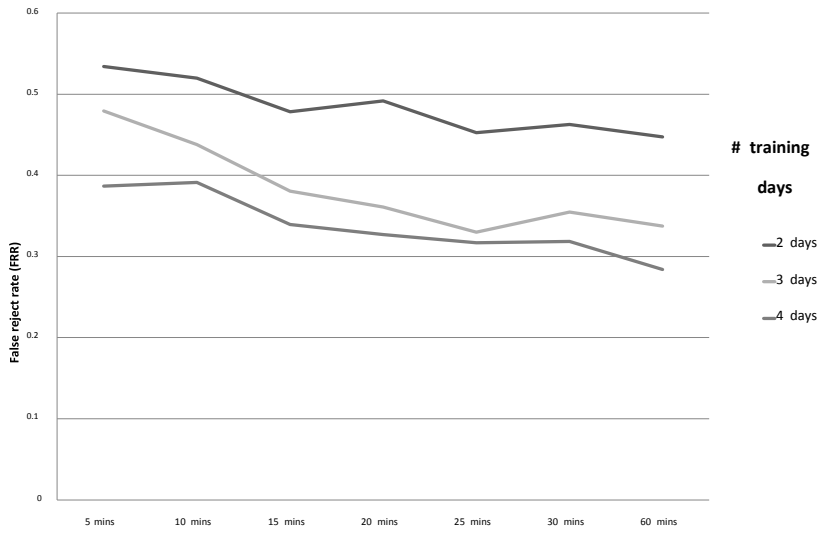


Figure 8: Percentage of remaining windows out of the total windows

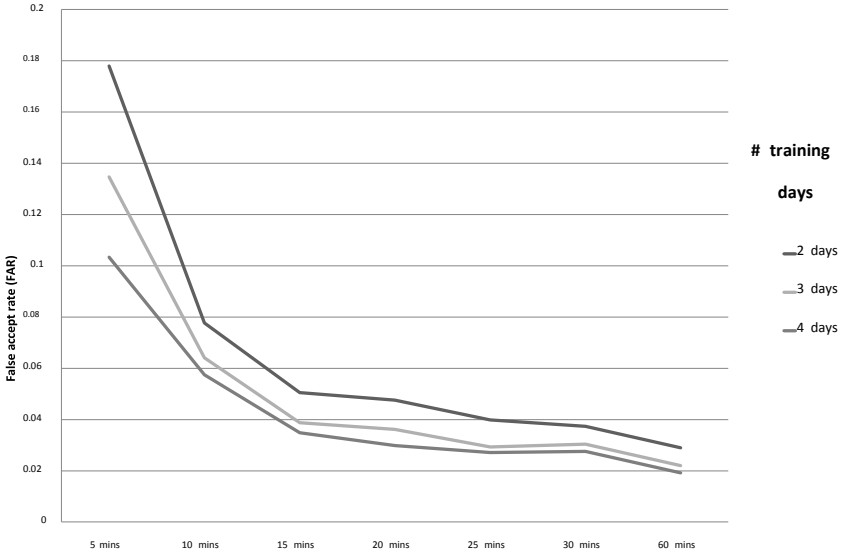
As expected, it can be seen that as the window size increases in both time and character count, the FRR decreases. The FAR shows a slightly different behavior: for each minimum characters-per-window threshold, there is still a decrease in FAR as the window time increases; however, as that character threshold increases, the FAR increases, especially shown with 5-minute windows. This can be caused by the increase in sparsity of the data, since as the characters threshold increases, less training data is available. At the same time, the percentage of remaining windows decreases, where in some instances almost no data is available.

For evaluating the fusion system (Section 3.3) we train each of the sensors on 4 out of the 5 days available for each user and test on the remaining day. However, this amount of data is rather limited, and in a real active authentication system it is expected that the classifiers will be trained on a substantial amount of data, based on weeks or more. Due to the limits of the collected dataset, 10-folds cross-validation was chosen for evaluation as it better illustrates the expected accuracy of the sensor in a real system. In order to illustrate the effect of the amount of data used for training, within the limits of our dataset, we ran a set of experiments where we trained the stylometry sensor on a 2, 3 and 4 of the days and tested on one of the remaining days. For each number of chosen training days, all combinations were tested (e.g. for 3 training days with one test day there are $2\binom{5}{3} = 20$ experiments). Averaged FRR and FAR results for training the stylometry sensor with a threshold of minimum 1000 characters-per-window on 2, 3 and 4 of the days and testing one of the remaining days are shown in Figure 9 and Figure 10, respectively. It can be seen that as the size of the training set increases, both FRR and FAR decrease. Similar results were obtained for the other minimum character-per-window thresholds.



Window size

Figure 9: Averaged false reject rate (FRR) for training



Window size

Figure 10: Averaged false accept rate (FAR) for training

4.2 Keyboard Dynamics and Mouse Movement

The training and testing for the low-level metrics of mouse and keyboard is similar to that described in section 4.1, except that we train each classifier on 3 days of data and use the fourth to set the FAR and FRR parameters for the classifier that are needed for the fusion algorithm. The fifth day is used for testing.

Figure 11 and Figure 12 show the FAR and FRR rates respectively for the two keystroke dynamics sensors K1 and K2 as the size of the decision window increases from 30 seconds to 10 minutes. Figure 13 and Figure 14 show the FAR and FRR rates respectively for the three mouse dynamics sensors M1, M2 and M3 as the size of the decision window increases from 30 seconds to 10 minutes. For all four figures, the performance is averaged over the 19 users and characterized with respect to time window size used by each of the sensors. Any data older than the duration of the window is discounted to zero by the sensors. The sensor only provides a decision when the time-window includes a minimum amount of feature events. For both mouse and keyboard that threshold is set to 10 events.

As the size of the decision window increases, the FAR and FRR rates generally decrease for all sensors.

For the mouse dynamics, we achieve similar levels of performance as the point-and-click metrics from [32] that our feature selection was based on. We require a larger time-window to achieve those error rates, but it's usable more frequently during the day's tasks, because we remove the constraint that a mouse movement must end in a click. An intuitive way to understand the difference between the two types of metrics is that a user in our dataset clicks the mouse an average of 1,938 times a day, but moves the mouse 106,733 times a day as Table 1 shows.

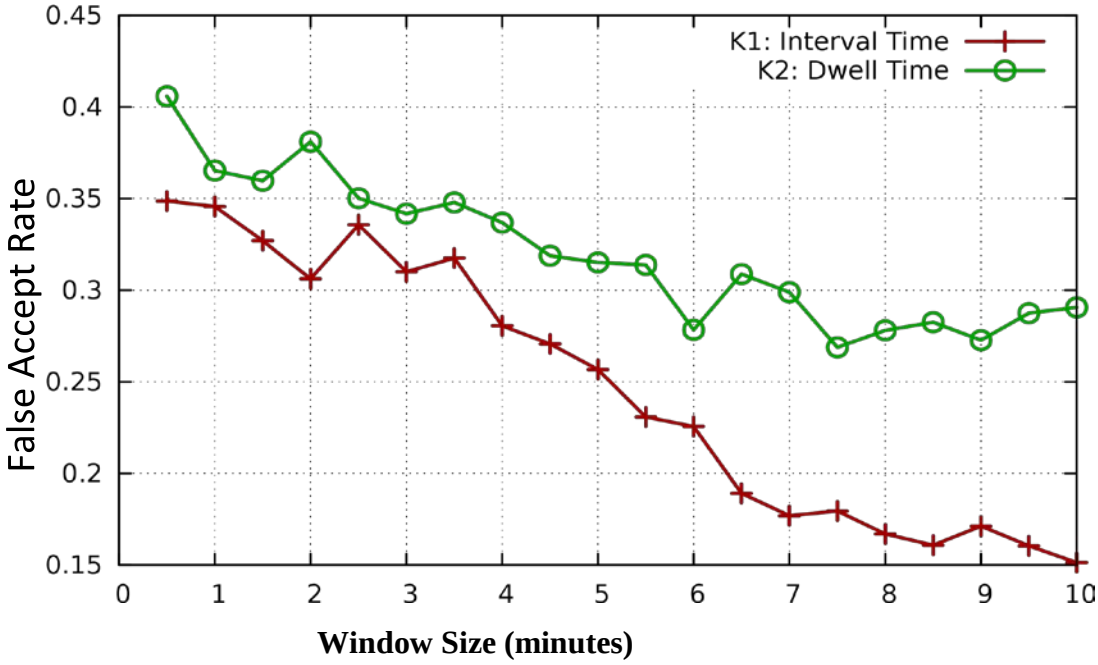


Figure 11: False accept rate (FAR) of the two individual keystroke sensors K1 & K2

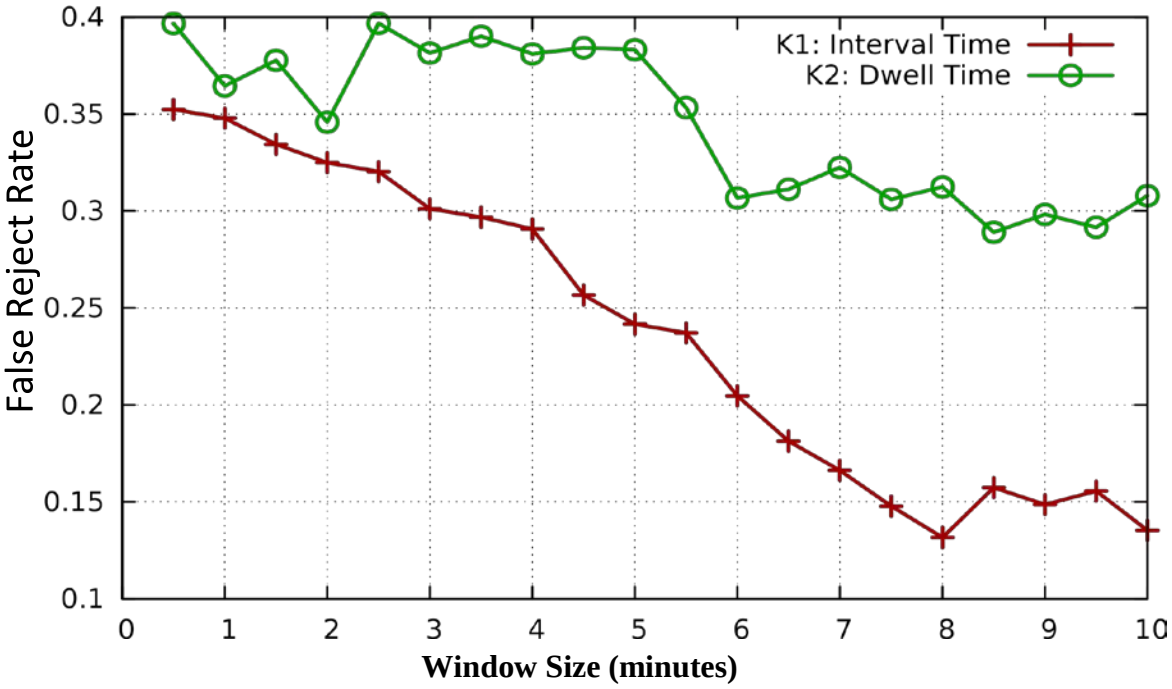


Figure 12: False reject rate (FRR) of the two individual keystroke sensors K1 and K2

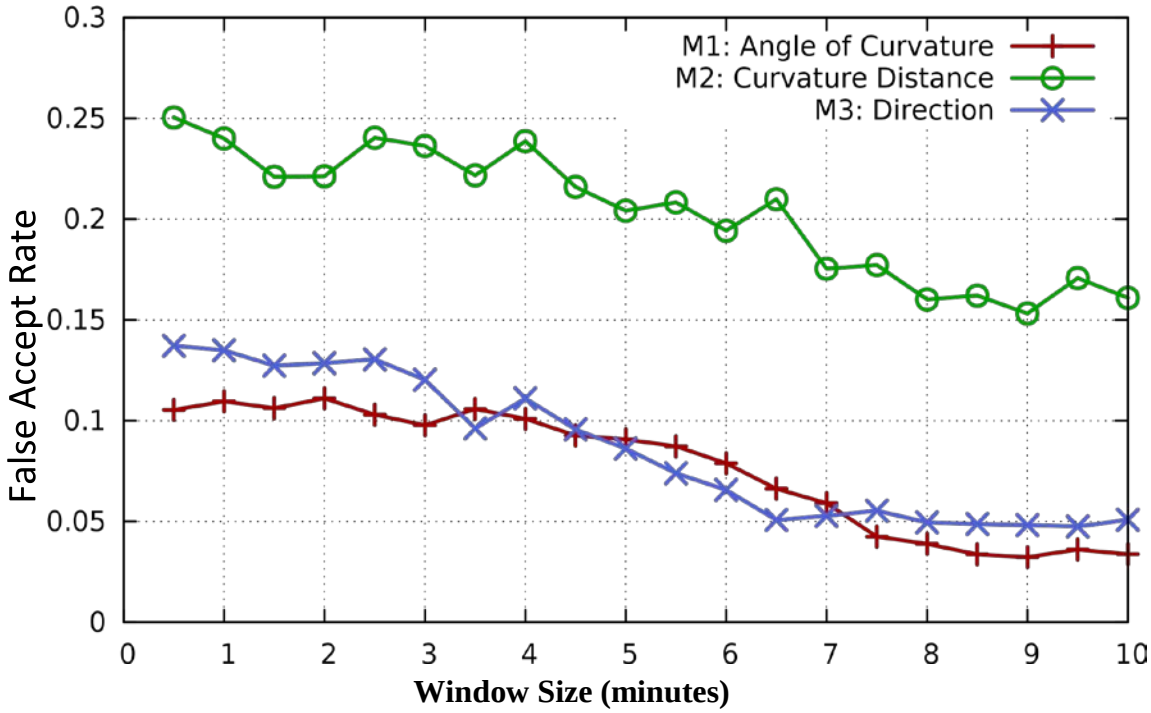


Figure 13: False accept rate (FAR) of the three individual mouse sensors M1,M2 & M3

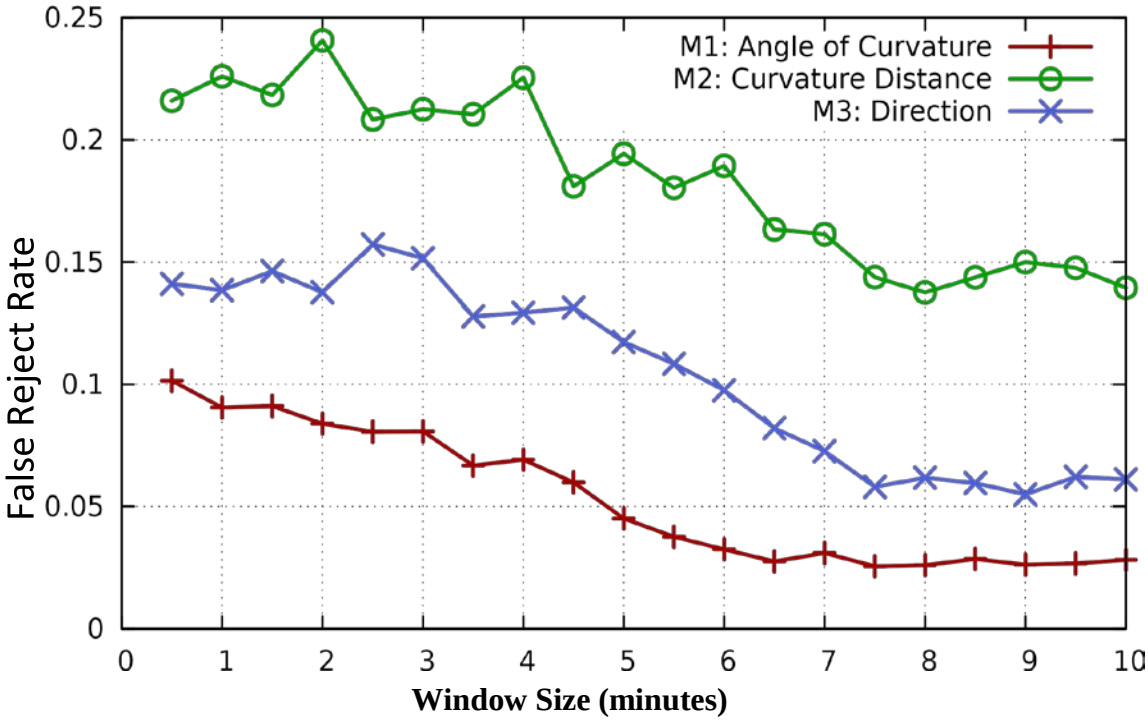


Figure 14: false reject rate (FRR) of the three individual mouse sensors M1, M2 & M3

4.3 Fusion of Low-Level Modalities

Figure 15 and Figure 16 show the FAR and FRR rates, respectively, achieved from the application of the fusion rule from section 3.3.1 on the 5 low-level metrics described in section 3.2.2. In both Figures, we plot the error rates of the individual sensors, and the error rates of the system that fuses them together. Once again, we characterize the increase in performance of the fusion algorithm as the size of the decision window increases from 30 seconds to 10 minutes. The fused sensors achieve lower average FAR and FRR rates than any of the sensors on their own.

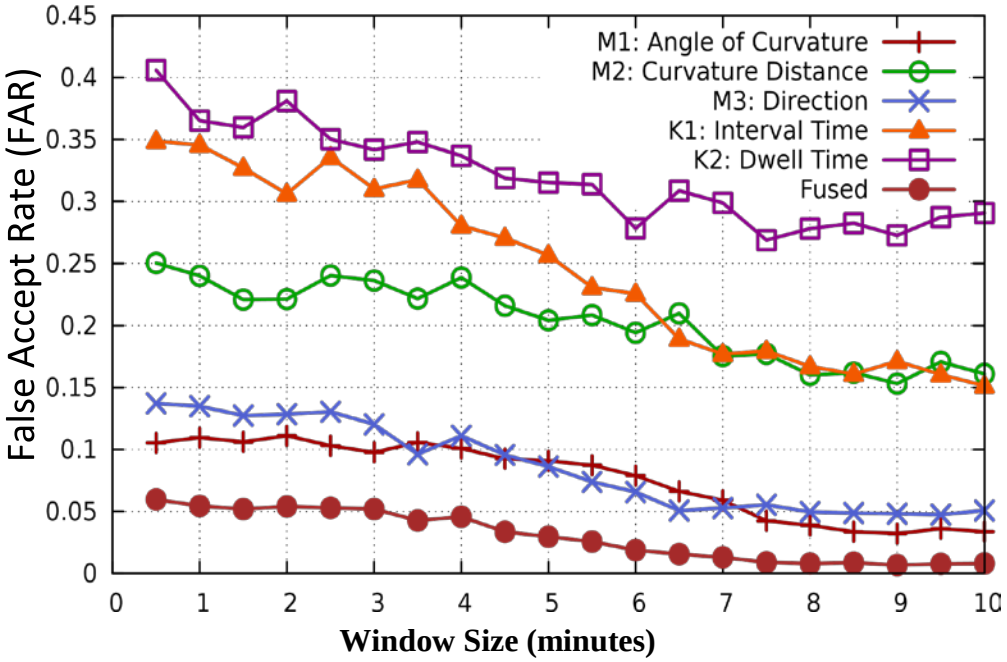


Figure 15: fused and individual false accept rate (FAR) of the five low-level sensors

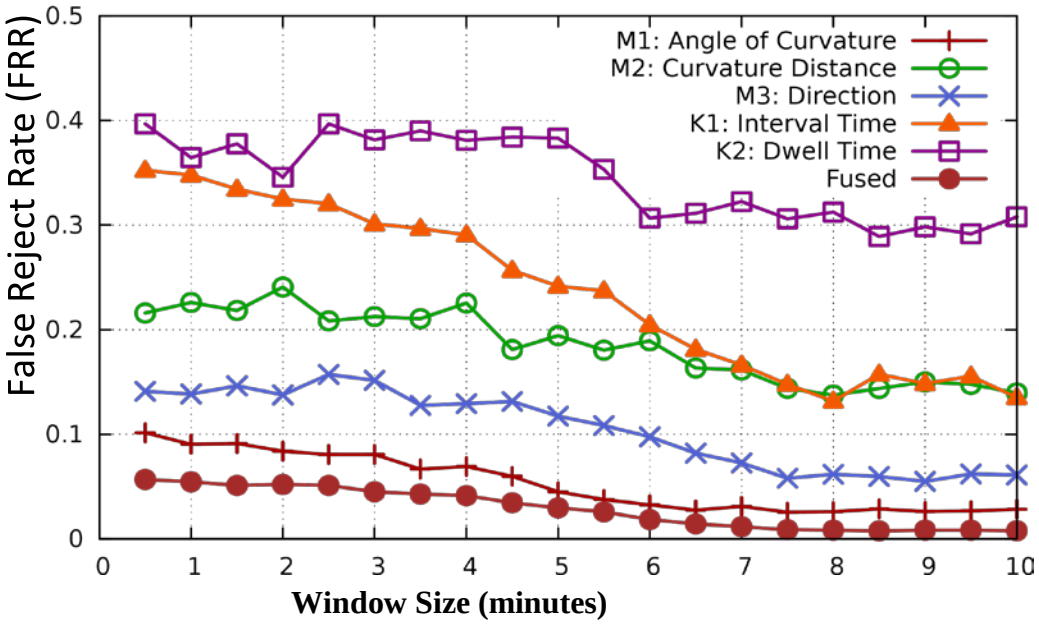


Figure 16: fused and individual false reject rate (FRR) of the five low-level sensors

4.4 Fusion of Low-Level and High-Level Modalities

For the next phase of fusion, we incorporated the stylometry-based and web browsing sensors with those introduced in the previous section. Using stylometry introduces high level analysis and enables to profile users with rich linguistic parameters along with the other characteristics captured by the other sensors. Since the matrix of stylometry sensor configurations discussed in section 3.2.1 is very large, we chose four configurations that express different points along the tradeoff between size of the windows (in time and in characters), and the performance – FRR/FAR and availability under the size constraints. Parameters and statistics of the chosen configurations are detailed in Table 6.

Table 6: Parameters & statistics for the stylometry sensor configurations

ID	Win. size	Min. chars	FRR	FAR	Availability
S1	30 minutes	1000	0.31861	0.02757	32.60%
S2	30 minutes	500	0.33827	0.02268	50.69%
S3	10 minutes	400	0.38915	0.02962	25.71%
S4	10 minutes	100	0.49113	0.03121	47.96%

The false reject and false accept rates (FRR and FAR) are evaluated using stylometry as a standalone sensor, averaged over results of training on 4 days and testing on the remaining day; The availability is the percentage of remaining windows that pass the minimum characters-per-window threshold.

The first two 30-minute-window configurations were chosen in order to allow large windows for text to be captured, yet they produce twice the data of the 60-minute windows configuration. Between these two, S1 refines the data even more, and by that decreases FRR on the expense of availability (only 32.6% of the windows are left after filtering out those with less than 1000 characters). S2 raises availability to a little over 50%, with only a slight increase in FRR (and a slightly better FAR). The other two 10minute-window configurations were chosen for their 3-times quicker response (i.e. decision output rate), a key parameter in an active authentication system. Although potentially less text is captured, the negative effect on FRR is counteracted to some extent by the increase in the number of windows – twice as many more than the first two configurations. Similarly to S1, S3 was chosen to refine the quality of the data, yet maintains a little over 25% availability for this time-wise short window configuration. Lastly, S4 was chosen as it is on the edge of the tradeoff, being the least-demanding configuration in terms of size, but quick in response and reasonably available (almost 50% like his 30-minute parallel, S2).

Table 7 shows the FAR and FRR for the fusion of the five low-level sensors together with all 32 combinations of the 5 high-level sensors all operating on a 10 minute window (except S1 and S2 operating on a 30 minute window). The tables are in ascending order according to their respective error rates. Each of the rows in the table represents one complete experiment with the checkmarks indicating whether a sensor is providing the fusion center with local decisions for that experiment. At the top in both cases is the system where all sensors are fused and at the bottom is the system where none of high-level sensors are fused. We can conclude from these results that in majority of cases, adding extra sensors decreases both FAR and FRR.

Table 7: FAR and FRR for the fusion of the five low-level sensors

W1	S1	S2	S3	S4	M1	M2	M3	K1	K2	FRR
X	X	X	X	X	X	X	X	X	X	0.00218
X	X	X		X	X	X	X	X	X	0.0023
X	X	X	X		X	X	X	X	X	0.0024
X	X		X	X	X	X	X	X	X	0.00244
	X	X	X		X	X	X	X	X	0.00248
	X	X	X	X	X	X	X	X	X	0.0027
X		X	X	X	X	X	X	X	X	0.0028
X	X	X			X	X	X	X	X	0.00282
	X		X	X	X	X	X	X	X	0.00336
X	X		X		X	X	X	X	X	0.0035
X		X	X		X	X	X	X	X	0.00352
X		X		X	X	X	X	X	X	0.00352
X			X	X	X	X	X	X	X	0.00362
	X	X		X	X	X	X	X	X	0.00364
		X	X	X	X	X	X	X	X	0.0039
X	X			X	X	X	X	X	X	0.00398
X		X			X	X	X	X	X	0.00402
X	X				X	X	X	X	X	0.00402
X			X		X	X	X	X	X	0.00442
	X	X			X	X	X	X	X	0.0048
		X		X	X	X	X	X	X	0.0049
	X			X	X	X	X	X	X	0.00526
	X		X		X	X	X	X	X	0.0054
		X			X	X	X	X	X	0.00566
			X	X	X	X	X	X	X	0.0057
			X		X	X	X	X	X	0.00574
X				X	X	X	X	X	X	0.00586
		X	X		X	X	X	X	X	0.00592
	X				X	X	X	X	X	0.00632
X					X	X	X	X	X	0.00714
				X	X	X	X	X	X	0.00746
					X	X	X	X	X	0.01016

W1	S1	S2	S3	S4	M1	M2	M3	K1	K2	FAR
X	X	X	X	X	X	X	X	X	X	0.00122
	X	X	X	X	X	X	X	X	X	0.0021
X	X		X	X	X	X	X	X	X	0.00218
X	X	X		X	X	X	X	X	X	0.00254
X	X	X	X		X	X	X	X	X	0.0026
X		X	X	X	X	X	X	X	X	0.00266
X		X	X		X	X	X	X	X	0.00278
X	X	X			X	X	X	X	X	0.00286
X	X		X		X	X	X	X	X	0.00288
	X		X	X	X	X	X	X	X	0.0029
X		X		X	X	X	X	X	X	0.0033
	X	X	X		X	X	X	X	X	0.00342
	X	X		X	X	X	X	X	X	0.00348
	X		X		X	X	X	X	X	0.00368
		X	X	X	X	X	X	X	X	0.00368
X			X	X	X	X	X	X	X	0.00378
		X	X		X	X	X	X	X	0.00384
	X	X			X	X	X	X	X	0.0039
X	X			X	X	X	X	X	X	0.004
X			X		X	X	X	X	X	0.00428
X	X				X	X	X	X	X	0.00442
X		X			X	X	X	X	X	0.00446
X				X	X	X	X	X	X	0.00476
		X		X	X	X	X	X	X	0.00512
			X	X	X	X	X	X	X	0.0057
	X			X	X	X	X	X	X	0.00592
				X	X	X	X	X	X	0.00652
			X		X	X	X	X	X	0.007
		X			X	X	X	X	X	0.00702
X					X	X	X	X	X	0.00728
	X				X	X	X	X	X	0.00738
					X	X	X	X	X	0.0102

4.5 Contribution of Individual Sensors and Robustness of Fusion System

In order to analyze the performance of the fusion system, we ran several experiments on a 67-user subset of the dataset. The 67 users were selected based on a threshold amount of data collected. Figure 17 is diagram of the 11 sensors used as part of the HCI suite in the experiments that determine the contribution of the HCI sensors.

The following are the experiments we ran, performance we observed, and the key insights we drew from those observations:

- **Extremely low closed-world error rates**: In Figure 18, we show the performance of the HCI suite fused with the two stylometry suites. The ROC curve is under the 0.01 error rate for both FAR and FRR which is one of the best biometrics-based detection rates we have seen in literature. Based on further experiments, we can conclude that this performance degrades significantly when a user outside the training set is classified.
- **Low dependence of number of users in the dataset**: In Figure 19, we show the FAR and FRR performance of the fusion system with a 10 user dataset and the 67 user dataset. We observe that the performance of the system degrades gradually with the number of users, but the degradation is sublinear.
- **Time to make binary classification**: In Figure 19, we also observe that whether a user is valid or not can be determined with less than 0.001 error rate in under 2 minutes of continuous activity.
- **Contribution of stylometry**: In Figure 20, we observe that when stylometry contributes to the gobal decision produced by the fusion center, its contribution is significant. The rate of decision contributiton is an order of magnitude lower, however, than that of the HCI sensors.
- **Contribution of HCI sensors**: In Figure 21, we observe that "mouse curve distance" is the sensor that contributes most (in performance and frequency) to the global decision produced by the fusion center. Largest contribution appear leftmost.

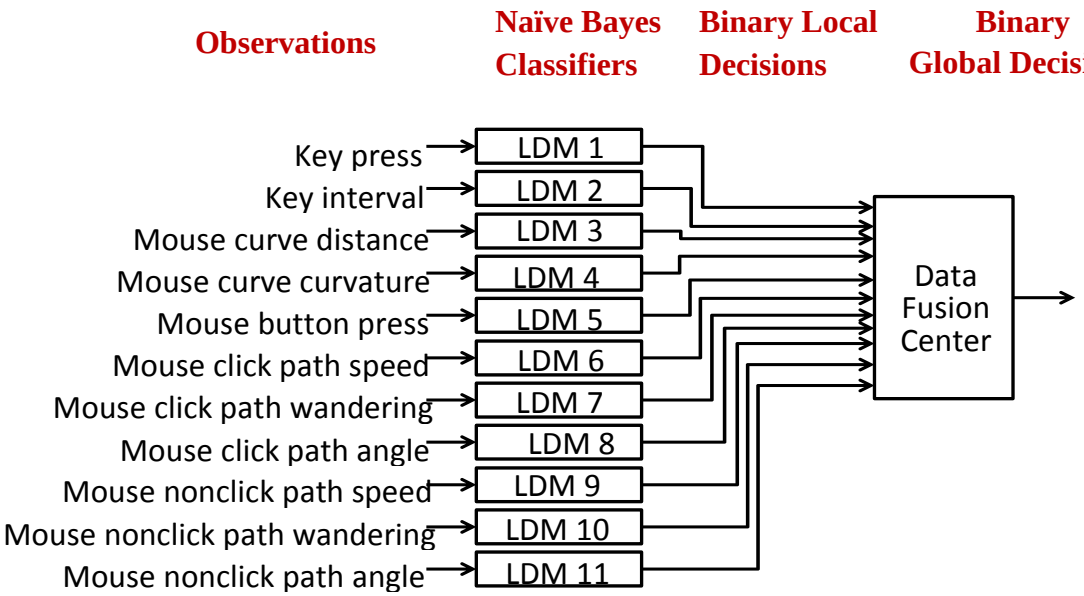


Figure 17: Diagram of the 11 sensors used as part of the HCI suite

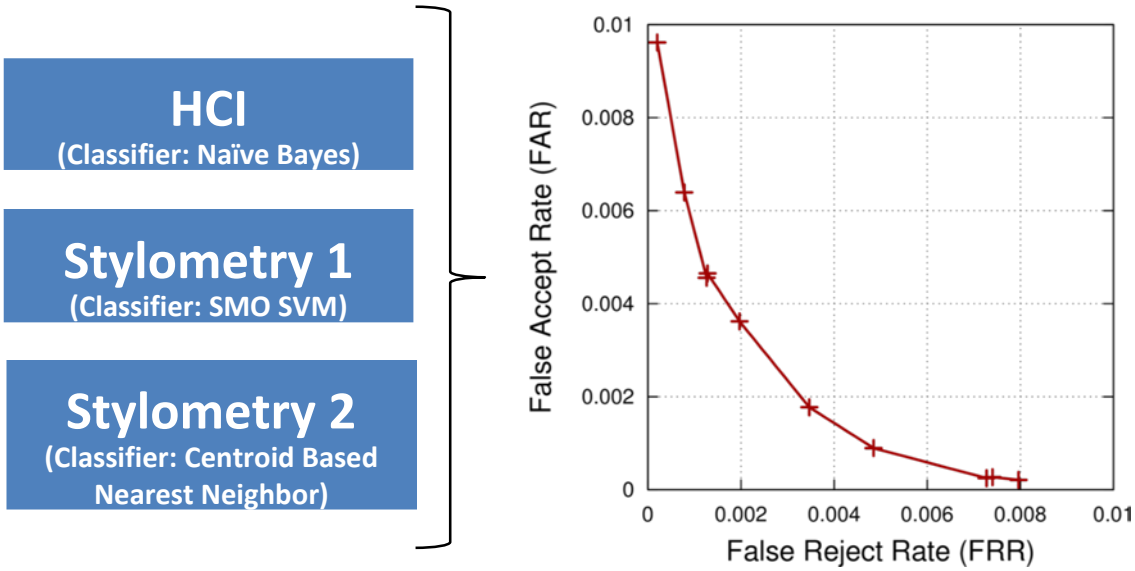


Figure 18: Performance of the fused system that incorporates HCI suite

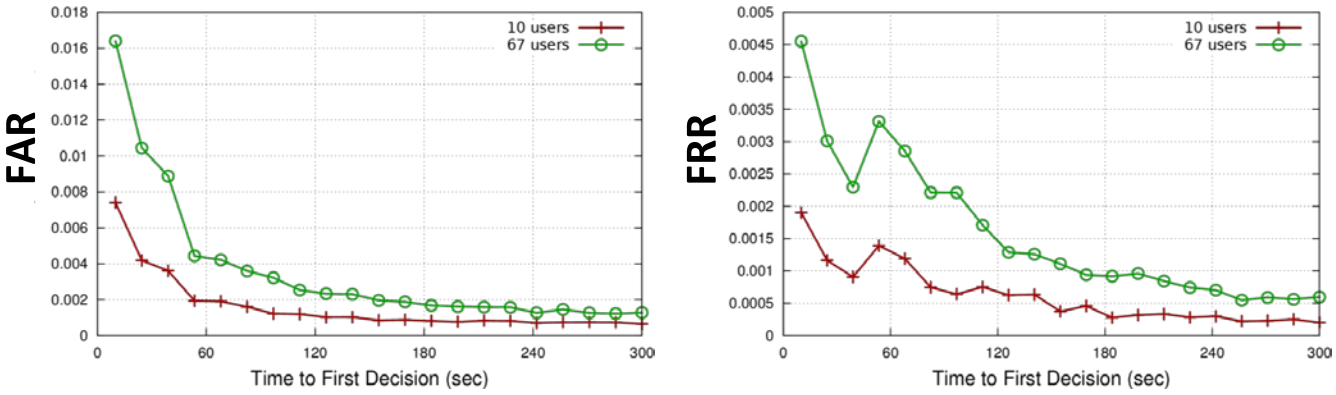


Figure 19: FAR and FRR performance of the fusion system

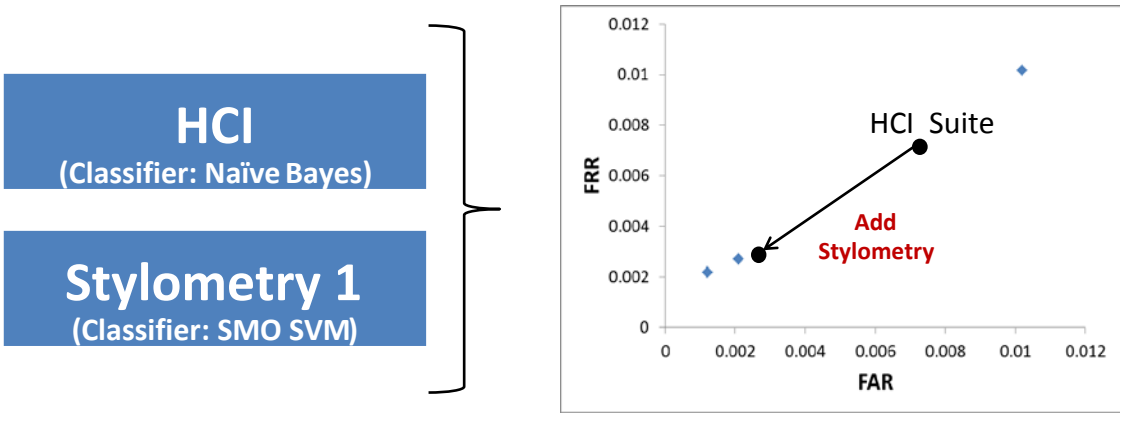


Figure 20: Performance of the fusion system for contribution of stylometric sensor

$$\text{Relative contribution} = \frac{\text{Pe(without sensor)– Pe(with sensor)}}{\text{Pe (without sensor)}}$$

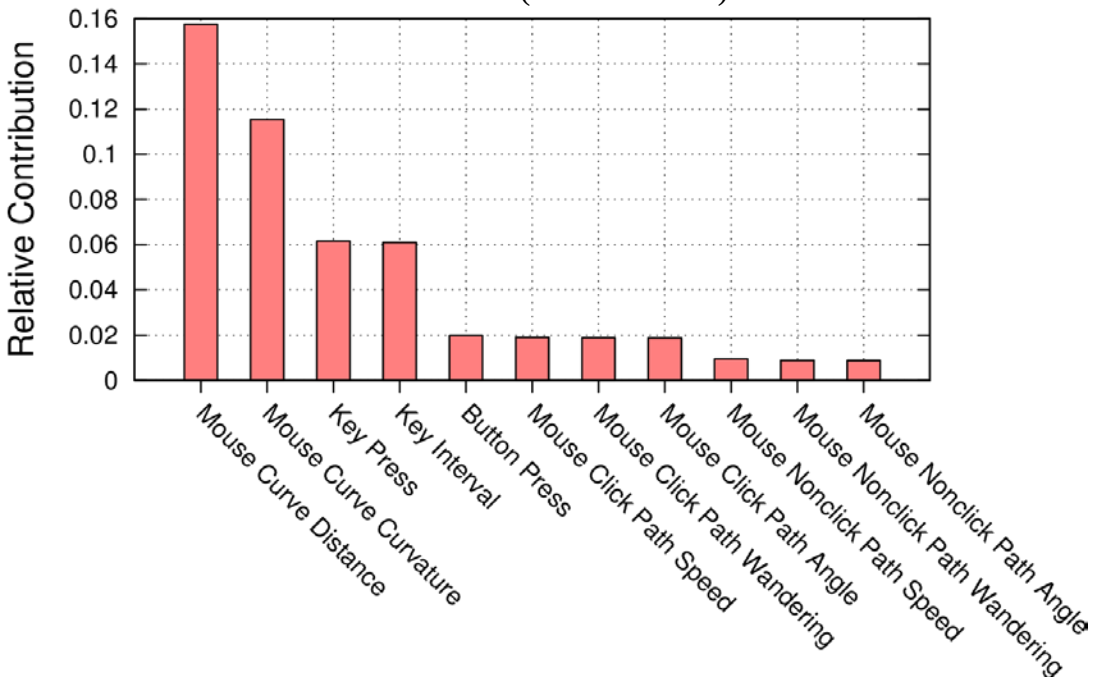


Figure 21: List of HCI sensors ordered by their contribution

- **Intruder detection**: Figure 22 shows an example scenario where an intruder enters at the 300 second mark and leaves at the 600 second mark. The fusion system accurately detects the intrusion in under 30 seconds. In Figure 22, we observe that when sensors operate under a 10 second window, the fused decision detects a change in user in under 30 seconds. This is a temporal perspective on the performance of active authentication that indicates the applicability of the system as a replacement for passwords in a closed-world environment such as the one considered in our work.
- **Robustness to adversarial users**: Figure 23 shows the performance of the fusion system based on the HCI suite when a number of the sensors are perfectly compromised by an adversary. In other words, a "compromised sensor" is one that produces results as if a legitimate user is at the computer. In Figure 23, we observe that the fusion system is robust to perfect compromise of 4 of the 11 sensors, but begins to break down upon further adversarial spoofing. In order to compromise more than four sensors, the adversarial user must perfectly mimic the keyboard dynamics and some of the aspects of the legitimate user's mouse movements.

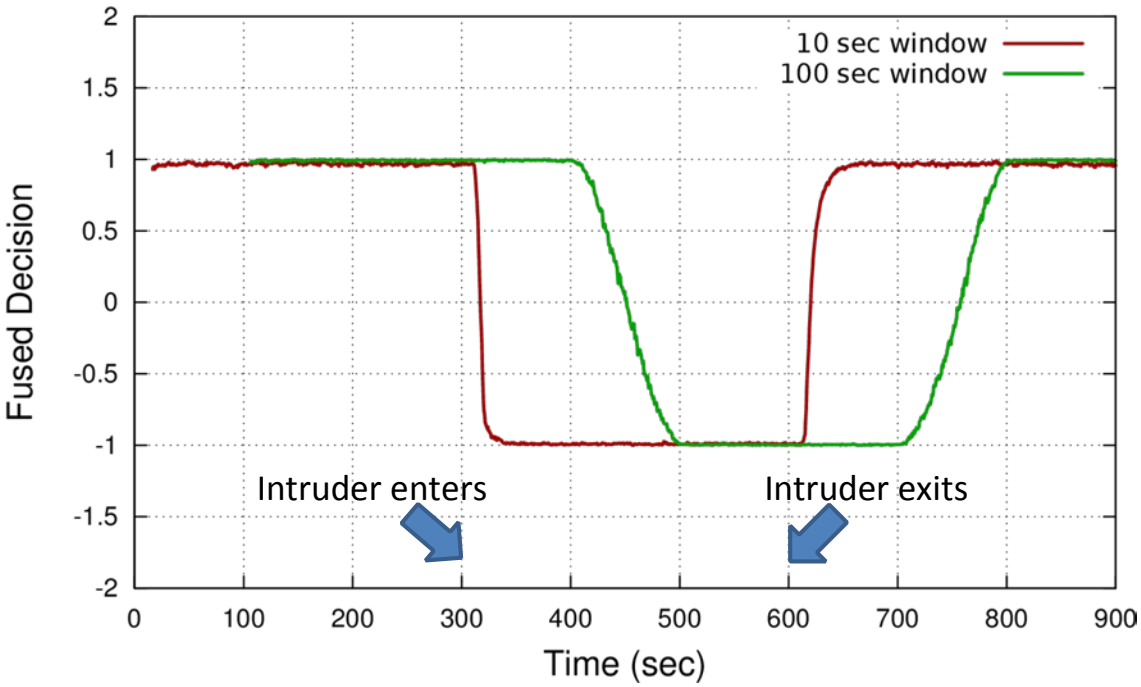


Figure 22: Example scenario

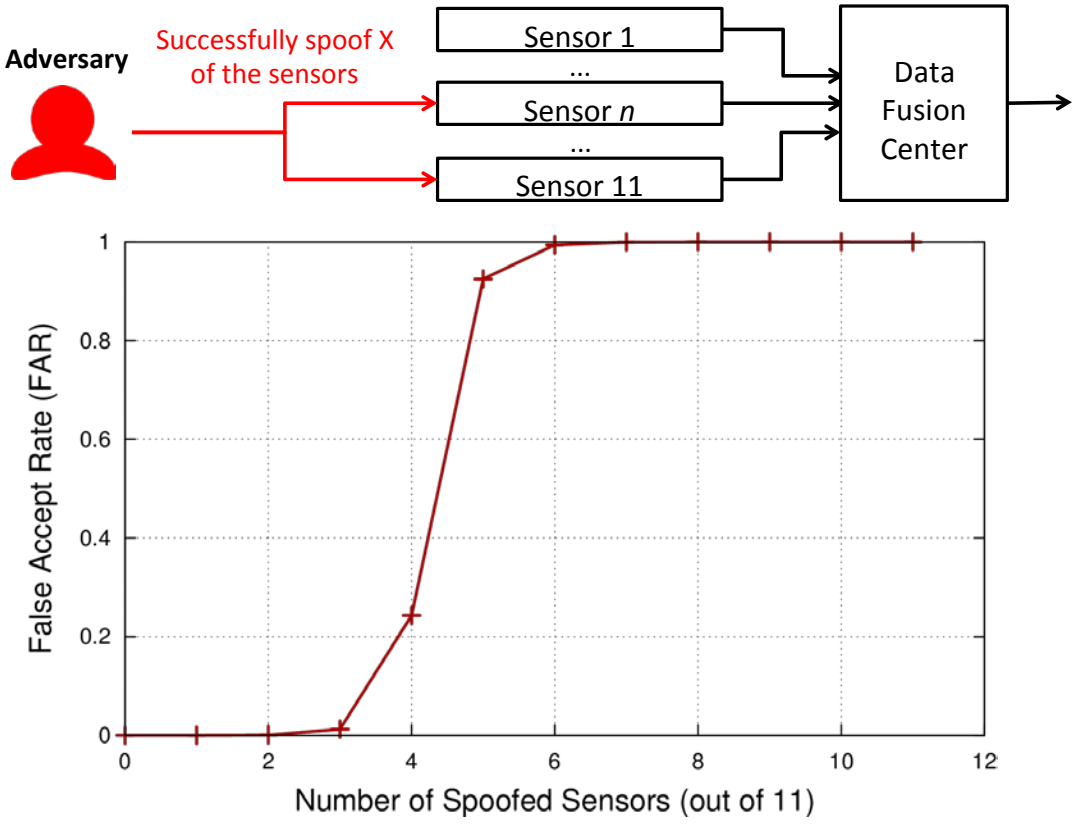


Figure 23: Performance of the fusion system based when compromised

5.0 CONCLUSION

In this report, we discuss a parallel binary detection decision fusion architecture for a representative collection of behavioral biometric sensors: keystroke dynamics, mouse movement, stylometry, web browsing behavior. Using this fusion method we address the problem of active authentication and characterize its performance on a dataset from a real-world office environment. The application of the Chair-Varshney fusion algorithm and the high-level sensors based on stylometry and web browsing behavior are novel in the active authentication context, and show promising performance in terms of low false acceptance rate (FAR) and low false rejection rate (FRR). We analyze the contribution of individual sensors, the time it takes to detect a change in user, and the robustness of the system to adversarial compromise of some of the sensors. Lastly, we look at alternative model of classification based on personality characteristics of the users.

6.0 REFERENCES

[1] A. Ahmed and I. Traore, "A new biometric technology based on mouse dynamics," *Dependable and Secure Computing, IEEE Transactions on*, vol. 4, no. 3, pp. 165 –179, july-sept. 2007.

[2] F. Bergadano, D. Gunetti, and C. Picardi, "User authentication through keystroke dynamics," *ACM Trans. Inf. Syst. Secur.*, vol. 5, no. 4, pp. 367–397, Nov. 2002.

[3] M. Obaidat and B. Sadoun, "Verification of computer users using keystroke dynamics," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 27, no. 2, pp. 261 –269, apr 1997.

[4] T. Ord and S. Furnell, "User authentication for keypad-based devices using keystroke analysis," in *Proceedings of the Second International Network Conference (INC-2000)*, 2000, pp. 263–272.

[5] L. JAMES, "Fundamentals of biometric authentication technologies," *International Journal of Image and Graphics*, vol. 1, no. 01, pp. 93–113, 2001.

[6] C. E. Chaski, "Who's at the keyboard: Authorship attribution in digital evidence invesigations," *International Journal of Digital Evidence*, vol. 4, no. 1, p. n/a, 2005, electronic-only journal: http://www.ijde.org, accessed 5.31.2007.

[7] ——, "The keyboard dilemma and forensic authorship attribution," Advances in Digital Forensics III, Boston, pp. 133–148, 2007.

[8] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, "Characterizing user sessions on youtube," *IEEE Multimedia Computing and Networking (MMCN)*, 2008.

[9] F. Benevenuto, T. Rodrigues, M. Cha, and V. Almeida, "Characterizing user behavior in online social networks," in *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, ser. IMC '09. New York, NY, USA: ACM, 2009, pp.49–62. [Online]. Available: http://doi.acm.org.ezproxy2.library.drexel.edu/10.1145/1644893.1644900

[10] R. Dogan, G. Murray, A. Nev´ eol,´ and Z. Lu, "Understanding pubmed R user search behavior through log analysis," *Database: the journal of biological databases and curation*, vol. 2009, 2009.

[11] J. Garcia-Dorado, A. Finamore, M. Mellia, M. Meo, and M. Munafo, "Characterization of isp traffic: Trends, user habits, and access technology impact," *Network and Service Management, IEEE Transactions on*, vol. PP, no. 99, pp. 1 –14, 2012.

[12] A. Ahmed and I. Traore, "A new biometric technology based on mouse dynamics," *Dependable and Secure Computing, IEEE Transactions on*, vol. 4, no. 3, pp. 165 –179, July-Sept. 2007.

[13] D. Shanmugapriya and G. Padmavathi, "A survey of biometric keystroke dynamics: Approaches, security and challenges," *Arxiv preprint arXiv:0910.0817*, 2009.

[14] J. M. and Carroll, "Creative names for personal files in an interactive computing environment," *International Journal of Man-Machine*

Studies, vol. 16, no. 4, pp. 405 – 438, 1982. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020737382800497 [15] H. Chapman, "The file naming habits of personal computer users," 1999.

[16] T. Sim, S. Zhang, R. Janakiraman, and S. Kumar, "Continuous verification using multimodal biometrics," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 4, pp. 687–700, 2007.

[17] J. Kittler, M. Hatef, R. Duin, and J. Matas, "On combining classifiers," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 3, pp. 226–239, 1998.

[18] S. Hashem and B. Schmeiser, "Improving model accuracy using optimal linear combinations of trained neural networks," *Neural Networks, IEEE Transactions on*, vol. 6, no. 3, pp. 792–794, 1995.

[19] Z. Chair and P. Varshney, "Optimal data fusion in multiple sensor detection systems," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. AES-22, no. 1, pp. 98 –101, jan. 1986.

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

[21] S. Argamon, M. Koppel, J. W. Pennebaker, and J. Schler, "Automatically profiling the author of an anonymous text," *CACM*, vol. 52, no. 2, pp. 119–123, February 2009.

[22] J. Rudman, "The state of authorship attribution studies: Some problems and solutions," *Computers and the Humanities*, vol. 31, pp. 351–365, 1998.

[23] P. Juola, "Authorship attribution," *Foundations and Trends in Information Retrieval*, vol. 1, no. 3, 2006.

[24] M. Koppel, J. Schler, and S. Argamon, "Computational methods in authorship attribution," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 1, pp. 9–26, 2009.

[25] E. Stamatatos, "A survey of modern authorship attribution methods," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 3, pp. 538–56, 2009.

[26] M. L. Jockers and D. Witten, "A comparative study of machine learning methods for authorship attribution," *LLC*, vol. 25, no. 2, pp. 215–23, 2010.

[27] H. van Halteren, R. H. Baayen, F. Tweedie, M. Haverkort, and A. Neijt, "New machine learning methods demonstrate the existence of a human stylome," *Journal of Quantitative Linguistics*, vol. 12, no. 1, pp. 65–77, 2005.

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

[29] J. N. G. Binongo, "Who wrote the 15th book of Oz? an application of multivariate analysis to authorship attribution," *Chance*, vol. 16, no. 2, pp. 9–17, 2003.

[30] O. V. Kukushkina, A. A. Polikarpov, and D. V. Khmelev, "Using literal and grammatical statistics for authorship attribution," *Problemy Peredachi Informatii*, vol. 37, no. 2, pp. 96–198, 2001, translated in "Problems of Information Transmission," pp. 172–184.

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

[32] N. Zheng, A. Paloski, and H. Wang, "An efficient user verification system via mouse movements," in *Proceedings of the 18th ACM conference on Computer and communications security*, ser. CCS '11. New York, NY, USA: ACM, 2011, pp. 139–150. [Online].
Available: http://doi.acm.org/10.1145/2046707.2046725

[33] N. Bakelman, J. V. Monaco, S.-H. Cha, and C. C. Tappert, "Continual keystroke biometric authentication on short bursts of keyboard input," in *Proceedings of Student-Faculty Research Day, CSIS, Pace University*, 2012. [Online]. Available: http://csis.pace.edu/ctappert/srd2012/d1.pdf

[34] M. Koppel, J. 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.

[35] H. van Halteren, "Author verification by linguistic profiling: An exploration of the parameter space," *ACM Transactions on Speech and Language Processing*, vol. 4, p. n/a, 2007.

[36] C. Gray and P. Juola, "Personality identification through on-line text analysis," in *Chicago Colloquium on Digital Humanities and Computer Science*, Chicago, IL, November 2011.

[37] P. Juola, M. Ryan, and M. Mehok, "Geographically localizing tweets using stylometric analysis," in *Proceedings of the American Association of Corpus Linguistics 2011*, Atlanta, GA, 2011.

[38] M. Koppel and J. Schler, "Authorship verification as a one-class classification problem," in *Proceedings of the twenty-first international conference on Machine learning*, ser. ICML '04. New York, NY, USA: ACM, 2004, pp. 62–. [Online]. Available: http://doi.acm.org/10.1145/1015330.1015448

[39] A. Abbasi and H. Chen, "Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace," *ACM Trans. Inf. Syst.*, vol. 26, no. 2, pp. 7:1–7:29, Apr. 2008. [Online]. Available: http://doi.acm.org/10.1145/1344411.1344413

[40] A. W. E. McDonald, S. Afroz, A. Caliskan, A. Stolerman, and R. Greenstadt, "Use fewer instances of the letter "i": Toward writing style anonymization." in *Lecture Notes in Computer Science*, vol. 7384. Springer, 2012, pp. 299–318.

[41] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, Nov. 2009. [Online]. Available: http://doi.acm.org/10.1145/1656274.1656278

[42] J. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods - Support Vector Learning*, B. Schoelkopf, C. Burges, and A. Smola, Eds. MIT Press, 1998. [Online]. Available: http://research.microsoft.com/˜jplatt/smo.html

[43] A. Abbasi and H. Chen, "Identification and comparison of extremist-group web forum messages using authorship analysis," *IEEE Intelligent Systems*, vol. 20, no. 5, pp. 67–75, 2005.

[44] M. Koppel and J. Schler, "Ad-hoc authorship attribution competition approach outline," in *Ad-hoc Authorship Attribution Contest*, P. Juola, Ed. ACH/ALLC 2004, 2004.

[45] 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*, vol. 57, no. 3, pp. 378–393, 2006.

[46] M. Karnan, M. Akila, and N. Krishnaraj, "Biometric personal authentication using keystroke dynamics: A review," *Applied Soft Computing*, vol. 11, no. 2, pp. 1565–1573, 2011.

[47] 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*. IEEE, 2006, pp. 117–126.

[48] 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*. IEEE, 2009, pp. 172–179.

[49] D. Umphress and G. Williams, "Identity verification through keyboard characteristics," *International journal of man-machine studies*, vol. 23, no. 3, pp. 263–273, 1985.

[50] S. Bleha, J. Knopp, and M. 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*. ACM, 1992, pp. 863–866.

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

[52] P. Druzhkov, V. Erukhimov, N. Zolotykh, E. Kozinov, V. Kustikova, I. Meerov, and A. Polovinkin, "New object detection features in the opencv library," *Pattern Recognition and Image Analysis*, vol. 21, no. 3, pp. 384–386, 2011.

[53] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[54] C. Chang and C. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[55] R. Yampolskiy, "Behavioral modeling: an overview," *American Journal of Applied Sciences*, vol. 5, no. 5, pp. 496–503, 2008.

[56] A. Q. Morton, *Literary Detection: How to Prove Authorship and Fraud in Literature and Documents*. New York: Scribner's, 1978.

[57] D. I. Holmes, "Authorship attribution," *Computers and the Humanities*, vol. 28, no. 2, pp. 87–106, 1994.

[58] M. Koppel, S. Argamon, and A. R. Shimoni, "Automatically categorizing written texts by author gender," *Literary and Linguistic Computing*, vol. 17, no. 4, pp. 401–412, 2002, doi:10.1093/llc/17.4.401.

[59] S. Argamon, S. Dhawle, M. Koppel, and J. W. Pennebaker, "Lexical predictors of personality type," in *Proceedings of the Classification Society of North America Annual Meeting*, 2005. [Online]. Available: citeseer.ist.psu.edu/744868.html

[60] B. Yu, Q. Mei, and C. Zhai, "English usage comparison between native and non-native english speakers in academic writing," in *Proceedings of ACH/ALLC 2005*, Victoria, BC, Canada, 2005.

[61] R. R. Tenney and J. Nils R. Sandell, "Decision with distributed sensors," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-17, pp. 501–510, 1981.

[62] M. Kam, W. Chang, and Q. Zhu, "Hardware complexity of binary distributed detection systems with isolated local bayesian detectors," *IEEE Transactions on Systems Man and Cybernetics*, vol. 21, pp. 565–571, 1991.

[63] A. R. Reibman and L. Nolte, "Optimal detection and performance of distributed sensor systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-23, pp. 24–30, 1987.

7.0 LIST OF SYMBOLS, ABBREVIATIONS AND ACRONYMS

- ACH Association for Computers and the Humanities
- AES Aerospace and Electronic Systems
- AFRL Air Force Research Laboratory
- ALLC Association for Literary and Linguistic Computing
- CSV Comma Separated Variable
- CTS Collaborative Technologies and Systems
- DFC Decision Fusion Center
- DOD Department Of Defense
- EER Equal Error Rate
- FAR False Acceptance Rate
- FRR False Rejection Rate
- HCI Human-Computer Interaction
- ICML International Conference on Machine Learning
- IEEE Institute of Electrical and Electronics Engineers
- LDM Local Decision Maker
- MIDAS Multiple Intelligences Developmental Assessment Scales
- MBTI Myers-Briggs Personality Inventory
- MMCN Multimedia Computing and Networking
- NEO Neuroticism-Extroversion-Openness
- NEO PI-R NEO Personality Inventory Revised
- SIGAPP Special Interest Group on Applied Computing
- TIST Transactions on Intelligent Systems and Technology
- USAF United States Air Force