WristPrint: Characterizing User Re-identification Risks from Wrist-worn Accelerometry Data

Nazir Saleheen
University of Memphis
nsleheen@memphis.edu

Md Azim Ullah
University of Memphis
mullah@memphis.edu

Supriyo Chakraborty
IBM T. J. Watson Research Center
supriyo@us.ibm.com

Deniz S. Ones
University of Minnesota
onesx001@umn.edu

Mani Srivastava
University of California, Los Angeles
mbs@ucla.edu

Santosh Kumar
University of Memphis
skumar4@memphis.edu

ABSTRACT

Public release of wrist-worn motion sensor data is growing. They enable and accelerate research in developing new algorithms to passively track daily activities, resulting in improved health and wellness utilities of smartwatches and activity trackers. But, when combined with sensitive attribute inference attack and linkage attack via re-identification of the same user in multiple datasets, undisclosed sensitive attributes can be revealed to unintended organizations with potentially adverse consequences for unsuspecting data contributing users. To guide both users and data collecting researchers, we characterize the re-identification risks inherent in motion sensor data collected from wrist-worn devices in users' natural environment. For this purpose, we use an open-set formulation, train a deep learning architecture with a new loss function, and apply our model to a new data set consisting of 10 weeks of daily sensor wearing by 353 users. We find that re-identification risk increases with an increase in the activity intensity. On average, such risk is 96% for a user when sharing a full day of sensor data.

CCS CONCEPTS

• **Security and privacy** → **Usability in security and privacy**; **Social aspects of security and privacy**; • **Human-centered computing** → **Ubiquitous and mobile devices**;

KEYWORDS

Privacy, User Re-identification, Wrist-worn Accelerometers

ACM Reference Format:
Nazir Saleheen, Md Azim Ullah, Supriyo Chakraborty, Deniz S. Ones, Mani Srivastava, and Santosh Kumar. 2021. WristPrint: Characterizing User Re-identification Risks from Wrist-worn Accelerometry Data. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS '21), November 15–19, 2021, Virtual Event, Republic of Korea.* ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3460120.3484799

CCS '21, November 15–19, 2021, Virtual Event, Republic of Korea

ACM ISBN 978-1-4503-8454-4/21/11…$15.00
https://doi.org/10.1145/3460120.3484799

1 INTRODUCTION

Consider the following two seemingly contrasting trends. First, a growing number of sensory datasets, such as mORAL [4], ExtraSensory [70], WISDM [72], Tesserae [47], and RAAMP2 [56], consisting of motion data from wrist-worn devices are being publicly released for research. They range from data collected in scripted settings [19, 45, 72] to data collected for days or weeks in the natural field environment [4, 47, 56]. It indicates a growing utility and adoption of wrist-worn devices (e.g., smartwatches, activity trackers), as well as a growing body of research that seeks to further improve the utility of these devices by developing algorithms to make new inferences of daily behaviors. These novel inferences include routine behaviors such as eating [68], drinking [6], brushing and flossing [3, 4], and potentially sensitive ones such as smoking [5, 51, 58], tremors [57], and pain [48].

Second, while the publicly released data is usually stripped of explicit identifiers and anonymized using recommended practices (e.g., using k-anonymity[67], l-diversity[44], and t-closeness[42]), there exists a growing body of inference attacks showing that protected attributes such as *age, gender, race*, and even *job type* can be inferred from accelerometry data alone [13, 17, 35, 75]. Even *user re-identification attacks* are shown to be feasible when available datasets are correlated with appropriately selected auxiliary data (e.g., restaurant check-ins) [31, 33]. *To improve the privacy protection of data contributors, both the users and the study researchers publicly releasing such data need a better understanding of the extent of re-identification risks embedded in wrist-worn motion sensor data under different data collection scenarios.*

In this paper, we analyze the re-identification risks from sharing wrist-worn accelerometry data collected in an unscripted, natural setting. Re-identification attack using (commonly shared) accelerometry data alone is significant, as it implies that a user contributing to different datasets can be linked, resulting in collective revelation of attributes, health states, and behaviors present in any of these datasets. For example, an insurance company or an employer collecting motion sensor data of its subscriber or employee (to reward healthy lifestyle) can learn of the users' prior history with smoking, pain, drug use, tremor, etc., from public datasets that this user may have contributed to previously (to advance science).

In our problem formulation, we assume an adversary has access to an anonymized sensor database consisting of labeled wrist-worn accelerometry data from n users. The labels may refer to a health condition or unhealthy daily behavior of the user that the researchers were seeking to develop a treatment or intervention for.

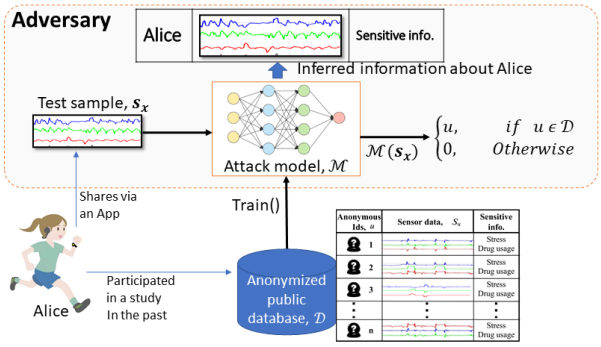


Figure 1: User re-identification attack from wrist-worn accelerometry traces.

Furthermore, the adversary also has access to wrist-worn accelerometry data from a user whose identity is known to the attacker (Figure 1). The goal of the adversary is to determine, with high confidence, if the user's data are also contained in the anonymized database and, if so, re-identify the anonymized user in the database.

To characterize re-identification risks from sharing wrist-worn accelerometer data, we undertake the following tasks. First, we formulate the re-identification problem as an open-set problem for greater generalizability. Second, we present a re-identification model, called *WristPrint*. It is composed of a base deep learning model architecture that combines a convolutional neural network (CNN) to extract a latent representation of micro-movements and a recurrent neural network (RNN) layer to identify temporal pattern in a sequence of micro-movements. We evaluate two boosting models that uses the output of the base model on each unit of test data to further improve the re-identification performance.

Third, to solve the open-set re-identification problem, we propose a novel consistency-distinction (CD) loss function. It guides the learning of our base model to minimize the intra-class variation (for consistency in identifying a user) and maximize the inter-class distance in feature space (to amplify distinction among different users). Taken together, such a loss function helps achieve a high re-identification rate for known users, while leaving the feature space largely unencumbered so as to recognize the absence of an unknown user when presented with their test data.

Fourth, we use a new dataset consisting of 353 users (full-time employees in diverse industries with a wide variety of job functions) who wore a wrist-worn device daily for ten weeks. Using a public dataset consisting of scripted activities, we train an activity classification model and apply it to the natural-life job performance dataset to partition this dataset into various activity types. We then train and test our *WristPrint* re-identification model to analyze the re-identification risks inherent in data collected when performing different types of activities in the natural environment.

Finally, to study the generalizability of our re-identification model beyond our dataset, we perform an entropy-based analysis and representation-overlap analysis via ROC characterization of true matching rate (TMR) and false acceptance rate (FAR).

Key Findings: First, we find from our experiments that out of various daily activities users engage in, exercise carries the highest re-identification risk and stationary state the lowest risk. Second, we observe that releasing one day of wrist-worn accelerometer data for a user in our dataset poses an average re-identification risk of 96%. Third, from entropy-analysis, we observe saturation around 100 distinct users in the training dataset. Fourth, we find that for the common activity of walking, the differential entropy is 56 bits, equivalent to $\approx 7.2 \times 10^{16}$ users. Finally, in our experiments when the model is provided with test data of 60 minutes, it achieves a TMR of 94%, with a FAR of $\leq 1.75\%$.

2 PROBLEM FORMULATION

We begin by introducing notations (see Table 1) and some definitions we use throughout the paper before formalizing the re-identification problem as an open-set machine learning problem [29].

Notation	Meaning
$\mathcal{I}_A$	Set of anonymized user indices, with $\lvert\mathcal{I}_A\rvert = n$
$\mathcal{D}$	Database of anonymized sensor traces, $\mathcal{D} = \bigcup_{u \in \mathcal{I}_A} \{(s_u, u)\}$
$\mathcal{I}_K$	Set of known identities of users, with $\lvert\mathcal{I}_K\rvert = k$
$\hat{s}_x$	Sensor trace of a user $x \in \mathcal{I}_K$
$\mathcal{K} : \mathcal{I}_A \rightarrow \mathcal{I}_K$	Secret function that maps anonymized indices of users to their unique known identities
Δ	Common unit length of segmentation for s_u
s_u^i	i^{th} segment (of length Δ) from s_u , for $i \in \{1, 2, ..., \lfloor\lvert s_u\rvert/\Delta\rfloor\}$
$\mathbb{D}$	$\mathbb{D} = \bigcup_{u \in \mathcal{I}_A} \left\{(s_u^i, u)_{i=1}^{\lfloor\lvert s_u\rvert/\Delta\rfloor}\right\}$
$\mathcal{F}$	Feature space of sensor segment in $\mathbb{D}$
$\phi : \mathbb{D} \rightarrow \mathcal{F}$	Feature generator
$\varphi : \mathcal{F} \rightarrow [0,1]^n$	Likelihood from classifier
$0 \leq \mathcal{T} \leq 1$	Decision threshold over the likelihood to declare positive re-identification
$M_\Delta = (\phi, \varphi, \mathcal{T})$	Base classification model
$\mathcal{M}$	Boosting model using m outputs from the Base model

Table 1: Symbols and Notations

Definition 2.1 (Sensor Trace)**.** Let sensor data point from a user u at time t be $s_u(t) = (p_1(t), p_2(t), ..., p_d(t)) \in \mathbb{R}^d$, where $p_j(t)$ is a single scalar value along one of d dimensions. A *segment* $s_u(t_s, t_e)$ is a contiguous time-series of sensor data from time t_s to t_e, i.e., $s_u(t_s, t_e) = \{s_u(t) : t_s \leq t \leq t_e\}$. Sensor trace s_u is a collection of all data segments from user u, i.e., $s_u = \bigcup_{(t_s,t_e)} \{s_u(t_s, t_e)\}$.

For our work, we use 3-axis accelerometry data (i.e., $d = 3$) collected from wrist-worn devices, e.g., activity trackers.

Definition 2.2 (Re-identification)**.** Let $\mathcal{I}_A = \{1, 2, ..., n\}$ be the set of user indices with non-empty sensor trace. Then, the anonymized sensor database $\mathcal{D} = \bigcup_{u \in \mathcal{I}_A} \{(s_u, u)\}$. Let $\mathcal{I}_K$ be the set of user identities that are known to the adversary. The adversary also has access to sensor trace $\hat{s}_x$ for a known user $x \in \mathcal{I}_K$. There exists a secret function (unknown to the adversary) $\mathcal{K} : \mathcal{I}_A \rightarrow \mathcal{I}_K$ that maps anonymized indices of users to their unique known identities.

2.1 Attack Model

Over the years, several defenses have been proposed for protecting data privacy and user anonymity. These include anonymization strategies that sanitize data by stripping them of personally identifying information and other quasi-identifying attributes [42, 44, 67], perturbations such as adding noise [43, 59, 71], generating and releasing only synthetic data that match desired properties of the original data [52, 74], and using cryptographic constructions to securely compute functions over data, protecting both data confidentiality and privacy [54]. However, at the time of releasing raw data, various pragmatic constraints such as the need for maximizing future research potential, low tolerance of some applications to noisy data (e.g., health diagnostics depend on preservation of the signal morphology [59]), and even limitations in adopting privacy techniques (e.g., choosing appropriate values for the privacy parameters (ϵ, δ) when using differential privacy [21, 38]), has led to anonymization strategies being preferred over others [23, 27, 28].

Accordingly, we use the following setting for our re-identification attack. We assume that the adversary has access to an anonymized sensor database, $\mathcal{D}$. As shown in Figure 1, the metadata (e.g., name, age, gender) associated with each user trace is suitably anonymized in $\mathcal{D}$, whereas sensor traces are released with minimal or no changes (e.g., using the data release mechanisms in [19, 70, 72]) and a user is only identified with their corresponding data index $u \in \mathcal{I}_A$. The adversary also has access to a user's sensor trace $\hat{s}_x$ together with their known identity $x \in \mathcal{I}_K$. The goal of the adversary now is to perform a two-step re-identification attack: (i) determine whether the user, corresponding to the trace $\hat{s}_x$, is in $\mathcal{D}$ (membership inference); and (ii) if present, to also determine the index u corresponding to the user, i.e., $\mathcal{K}(u) = x$ (identity matching).

Akin to other re-identification problems [36, 37], our problem can be formulated as a similarity search problem. In a similarity search problem, one is given a database D of items and a similarity function. The similarity score is high if two items are similar, and low, otherwise. Given a new item, one wants to efficiently find the item closest to this new item in the database. Usually, in a similarity search problem, the similarity metric is defined using a suitable mapping $\phi : D \rightarrow F$ of the items in D to some metric space F, and then $sim(s_i, s_j) = \mathrm{d}_F(\phi(s_i), \phi(s_j))$, where $\mathrm{d}_F(\cdot, \cdot)$ denotes the metric in F. For example, in the fingerprint matching problem, the mapping might map a fingerprint image to fixed set minutiae [36] or a compact fixed length FingerCode [37], with the distance metric being the Euclidean distance.

To solve our user re-identification problem, one can find, using a suitable similarity search query, the most similar data to the given input $\hat{s}_x$. Here, our database $\mathcal{D}$ consists of time-series segments, which may correspond to different activity states, e.g., walking, exercising, stationary, etc. Our challenge is to identify suitable structural patterns that can be considered as features or latent space, which constitutes the mapping ϕ and a metric space F, in which the distance function for quantifying similarity is defined. We develop machine learning algorithms to discover discriminative features from these time series data segments.

We assume the adversary employs a classification model $\mathcal{M}$ for the re-identification task. An attack is successful if the attacker can correctly re-identify the user, i.e., if for any given input $\hat{s}_x$ from a known user $x \in \mathcal{I}_K$ who also contributed data in the database $\mathcal{D}$, the model $\mathcal{M}$ outputs user u such that $\mathcal{K}(u) = x$, and similarly, for any given input $\hat{s}_x$ from a known user who did not contribute to the database $\mathcal{D}$, $\mathcal{M}$ outputs 0 for "not present."

2.2 Privacy Risks

Let $\mathcal{D}$ be a released anonymized database and $\hat{s}_x$ the sensor trace of an arbitrary user whose identity is known to the adversary. We assume that the attacker learns a machine learning model $\mathcal{M}$ using $\mathcal{D}$ such that for any test data $\hat{s}_x$, $\mathcal{M}(\hat{s}_x)$ outputs the closest matching user identifier from the database, if the matching score is acceptable, and 0, otherwise, when the user is determined to not have any data in the database. The re-identification risk, $\mathcal{R}(\mathcal{D})$, is defined as the expected probability with which the model, $\mathcal{M}$, accurately predicts the index of user $u \in \mathcal{I}_A$, if the user has data in $\mathcal{D}$, i.e., $\mathcal{K}(u) = x$, and predicts 0, when there exists no u such that $\mathcal{K}(u) = x$. To formally define this risk, we define two metrics — the detection and identification rate (DIR) or true matching rate (TMR), and the false alarm rate (FAR) — which are often used to characterize the performance of open-set identification problems.

True Matching Rate (TMR). Sensor trace $\hat{s}_x$ from a known user $x \in \mathcal{I}_K$ is detected if $\mathcal{M}$ correctly identifies $u \in \mathcal{I}_A$, with $\mathcal{K}(u) = x$.

$$TMR \text{ or } DIR = \frac{|\{(\hat{s}_x, x) | \mathcal{M}(\hat{s}_x) = u, \text{ where } \mathcal{K}(u) = x\}|}{|\{(\hat{s}_x, x) | \exists u \in \mathcal{I}_A, \text{ such that } \mathcal{K}(u) = x\}|} \quad (1)$$

False Acceptance Rate (FAR). False acceptance occurs when $\mathcal{M}$ falsely detects a user index $u \in \mathcal{I}_A$ in the database $\mathcal{D}$ for a sensor trace $\hat{s}_x$ from a known user $x \in \mathcal{I}_K$ with no data in $\mathcal{D}$. Note, $\mathcal{M}$ should output 0 if no data for user x is present in $\mathcal{D}$.

$$FAR = \frac{|\{(s_x, x) | \mathcal{M}(s_x) = u \in \mathcal{I}_A, \text{ where } \mathcal{K}(u) \neq x\}|}{|\{(s_x, x) | \nexists u \in \mathcal{I}_A, \text{ such that } \mathcal{K}(u) = x\}|} \quad (2)$$

There is a trade-off between TMR and FAR that is usually shown on a receiver operator characteristic (ROC).

Let $\rho_x = Pr[\exists u \in \mathcal{I}_A : \mathcal{K}(u) = x]$ be the probability of user x having data in $\mathcal{D}$. Then, we express the expected re-identification risk ($\mathcal{R}(\mathcal{D})$) as follows, using the model's performance.

$$\begin{aligned} \mathcal{R}(\mathcal{D}) &= \mathbb{E}_{(\hat{s}_x, x)}\Big(Pr[\mathcal{M}(\hat{s}_x) = u | u \in \mathcal{I}_A, \mathcal{K}(u) = x] * \rho_x \\ &\quad + \big(1 - Pr[\mathcal{M}(\hat{s}_x) = u | u \in \mathcal{I}_A, \mathcal{K}(u) \neq x]\big) * (1 - \rho_x)\Big) \\ &= TMR * \rho_x + \big(1 - FAR\big) * (1 - \rho_x) \end{aligned} \quad (3)$$

Thus, $\mathcal{R}(\mathcal{D})$ is a weighted average of TMR and (1-FAR).

3 PROPOSED MODEL: *WRISTPRINT*

We now present the overall architecture of the attack model and a high-level overview of the re-identification attack.

Overview of the *WristPrint* Approach: An overview of the end-to-end *WristPrint* model appears in Figure 2. First, a given sensor trace is segmented into fixed-size units. Then, a feature generator maps these segments into a feature space. The goal of feature transformation is that all the points in the feature space from the same individual cluster together, and clusters of different users are maximally separated. The base classifier assigns each feature vector to the nearest user id. Finally, the outputs of the base classifier for each segment are aggregated to determine the best user id label. We

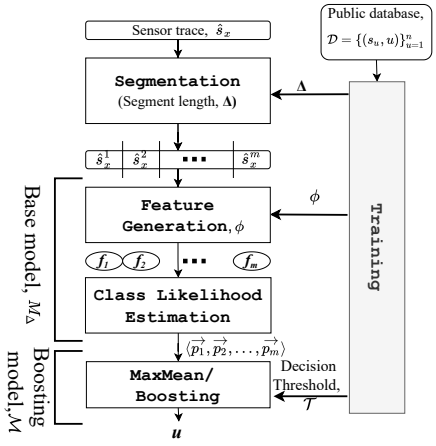


Figure 2: Overview of the *WristPrint* model.

Algorithm 1 `WristPrint`

Input: $\mathcal{D}$: Sensor dataset of n users
$\hat{s}_x$: Sensor trace from a test user
Output: User index of $\hat{s}_x$
function TrainBaseModel($\mathcal{D}, \Delta$)
  $\mathbb{D} = \{\texttt{Segment}(s_u, \Delta) | s_u \in \mathcal{D}\}$
  $M_\Delta \equiv (\phi, \varphi, \mathcal{T}) = \texttt{train}(\mathbb{D})$
    ▷ ϕ: Feature generator, φ: classifier, $\mathcal{T}$: decision threshold
  return M_Δ
end function
$M_\Delta, \Delta = \arg\max_\Delta \texttt{TrainBaseModel}(\mathcal{D}, \Delta)$
$\langle \hat{s}_x^1, ..., \hat{s}_x^m \rangle = \texttt{Segment}(\hat{s}_x, \Delta)$ ▷ $m = \left\lfloor \frac{|\hat{s}_x|}{\Delta} \right\rfloor$
$P \leftarrow \phi$ ▷ Initialize
for $i = 1$ **to** m **do**
  $x = M_\Delta(\hat{s}_x^i)$
  P.append(x)
end for
$u \leftarrow \mathcal{M}(P)$ ▷ Applying boosting method
return u

call this base-boosting pair model architecture since the base model takes each unit length segment as input and detects user identifiers as output, and the boosting model groups the user identifiers from the base model to produce a single detection.

Re-identification Attack Algorithm: Algorithm 1 describes different steps of our proposed re-identification algorithm. It takes a sensor database $\mathcal{D}$ of n users and a test sensor trace $\hat{s}_x$. At first, it trains the base model using the database, $\texttt{TrainBaseModel}(\mathcal{D})$ for different values of the unit length Δ. As described in Algorithm 1, it segments each sensor trace $s_u \in \mathcal{D}$ into unit segments of length Δ using $\texttt{Segment}(s_u, \Delta)$ function. Let $\mathcal{S}_u = \{s_u^1, s_u^2, ..., s_u^m\}$, with $m = \lfloor |s_u|/\Delta \rfloor$, be the set of all Δ-long segments of sensor data from user u. These segments from all users generate a new database $\mathbb{D} = \{(s_u^i, u) | s_u^i \in \mathcal{S}_u, \forall u \in \mathcal{I}_A\}$.

We train a base model M_Δ, using new database $\mathbb{D}$, such that it can assign a user id to each test sensor data segment $\hat{s}_x^i$ of length Δ. We assume M_Δ is a function composition of ϕ and φ, i.e., $M_\Delta = \phi \circ \varphi$. Here, the function ϕ is trained by a neural network model, which maps $s_u^i \in \mathbb{D}$ into an appropriate feature space $\mathcal{F}$. The function ϕ needs to be trained such that feature space $\mathcal{F}$ preserves consistency among features generated by data from the same user and distinction among features generated by data from different users. The goal is to maintain intra-class similarity (for same users) and inter-class differences (for different users) in the feature space.

To solve the similarity search problem, the attacker trains a classifier, $\varphi : \mathcal{F} \rightarrow [0, 1]^n$, that creates clusters in feature space for each class. For a feature vector of any given sensor data segment $\hat{s}_x^i$, it outputs the probability of each class, i.e., $\varphi(\phi(\hat{s}_x^i)) = \langle p_1, .., p_n \rangle$, where $p_j = Pr[u = j | \hat{s}_x^i]$ for anonymous index $1 \leq j \leq n$. Finally, a threshold $\mathcal{T}$ is learned, such that if all the probabilities are less than the threshold, then the model M_Δ, outputs "not present". We also find the optimal unit length Δ during the training process. We refer to M_Δ as the **Base Model** since it works on unit length data.

For a given database $\mathcal{D}$ and a sensor trace $\hat{s}_x$ of length more than Δ, the attacker creates a **Boosting Method** $\mathcal{M}$ by applying the base model $m = \lfloor |\hat{s}_x|/\Delta \rfloor$ times. The Boosting model combines the results of the base model to generate the final output. Boosting capability depends on the value of $l = |\hat{s}_x|$, segment length Δ, and the size of training data in $\mathcal{D}$.

4 THE BASE MODEL DESIGN

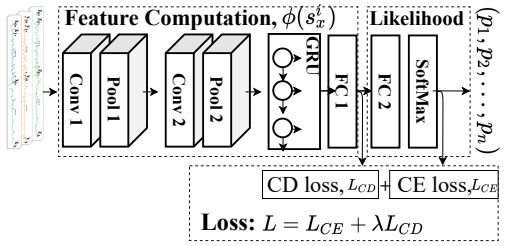


Figure 3: Convolutional-recurrent layers-based base model architecture. A combination of consistency-distinction (to minimize intra-class and maximize inter-class differences) and cross-entropy loss functions is used for model training.

We now present the full architecture of our proposed base model. To discover distinctive features for each person from unstructured accelerometry data, we develop a deep-learning architecture. The base model receives as input accelerometry trace (corresponding to specific activity states) segmented in units of length Δ. In Section 5.2, we describe how to find the best value of Δ.

Our base model consists of two major computational blocks: feature computation and class likelihood. Since we seek to identify unique pattern over the given accelerometry time-series, we consider the signal characteristics along both the time and amplitude axes to create a unique fingerprint of the user. As shown in Figure 3, base model's overall architecture consists of two convolutional blocks (convolutional layer followed by a max-pool layer), one recurrent layer, two fully connected layers, and a softmax layer. Accelerometry segments from wearable sensors are first processed

by the two convolutional blocks to learn micro features from the raw sensor data such as wrist movement or rotation. Next block in the pipeline is a Gated Recurrent Unit (GRU) to capture temporal patterns of the micro-feature sequence. Third block in the pipeline are the fully connected layers used to generate a classification score. Finally, the output of the fully connected layer is passed through a softmax function to generate the likelihood of each class.

Details regarding specific instantiation of these layers are as follows. We use 1-d convolution layers, each with 120 filters. The max-pool filters, following each convolution layer, are of size two. For regularization in the GRU layer, we user a dropout layer with a probability of 0.5. The final fully connected (dense) layer outputs a vector of dimension n, which is the number of classes (or users).

4.1 The Proposed Loss Function

Key to training a deep learning model for open set recognition is the choice of an appropriate loss function. We propose a new loss function that can guide the deep learning model to discover a representation of the input data and an accompanying classifier that can extract commonality among the data segments belonging to the same user and maximize distinction among the data segments from all other users (including the unseen ones). We now describe our loss function, which we call *consistency-distinction (CD) loss*.

Consistency is preserving the commonality of the signal from the same participant, and *distinction* is amplifying the differences among different participants. Both are essential for an open set recognition task [29]. We want to project raw accelerometry data to a feature space representation that the deep learning model can use to identify class boundaries satisfying both consistency and distinction. With the standard cross-entropy loss, our proposed architecture ensures separation among different users/classes, but it does not guarantee consistency and distinction.

Our CD loss function builds upon the commonly used Triplet Loss[60] and Center-Loss[73] functions. Triplet Loss seeks to maximize the separation among the classes (to amplify distinction), while Center Loss seeks to minimize the footprint of each class (to sharpen consistency). We first briefly introduce these two loss functions, and then describe our proposed CD loss function.

4.1.1 Triplet Loss. The triplet loss [10, 60] is usually trained on a series of triplets (s_u^i, s_u^j, s_v^k), where s_u^i and s_u^j are data from the same user u, and s_v^k is from a different user v. Triplet loss is designed to keep s_u^i closer to s_u^j than s_v^k, and widely used in many areas, such as face recognition and person re-identification [60]. It is formulated as follows:

$$\mathcal{L}_{trp} = \sum_{(s_u^i, s_u^j, s_v^k)} \{\|\phi(s_u^i) - \phi(s_u^j)\| - \|\phi(s_u^i) - \phi(s_v^k)\| + \alpha\},$$

where $\phi(s_u^i)$ denotes features from input s_u^i. Threshold α is a margin enforced between positive and negative pairs, ensuring that the minimum separation among different classes is at least α. The above formulation of triplet loss adopts Euclidean distance to measure the similarity of extracted features from two sensor segments.

4.1.2 Center Loss. For each iteration of training a deep learning model, Center Loss [73] to be used in the current iteration is trained on a mini-batch consisting of several data segments (s_u^j) from $\mathbb{D}$ of the same user u, i.e. $\mathbb{D}_{MB} \subset \mathbb{D}$. The collection of s_u^j are randomly selected from $\mathbb{D}$ so $\mathbb{D}_{MB}$ can consist of any data segment from any user. The Center Loss function seeks to minimize the intra-class variations. Using $\phi(\mathcal{S}_u)$ to denote deep features of all data segments from a user u, the Center Loss function is defined as

$$\mathcal{L}_C = \frac{1}{2} \sum_{s_x^j \in \mathbb{D}_{MB}} \|\phi(s_u^j) - \overline{(\phi(\mathcal{S}_u))}\|_2^2,$$

where $\overline{(\phi(\mathcal{S}_u))}$ is the centroid of deep features from Class u.

4.1.3 The Consistency-Distinction (CD) Loss Function. As described above, the Triplet Loss function can be used to maximize inter-class separation, and the Center Loss function can be used to maximize the intra-class consistency. But, our goal is to guide the deep learning model to achieve both distinction and consistency together. There are several challenges in developing a new loss function that can simultaneously optimize both criteria.

First, the inputs for both loss functions are different. The Triplet Loss function expects a triplet consisting of two data segments from the same user and the other data segment from another user in each training iteration. The Center Loss, on the other hand, expects a mini-batch randomly selected from all training data, without any preference for selecting data segments belonging to a common user. The second challenge is how to adapt the consistency metric so that the footprint of the classes are not disproportionately enlarged due to the presence of some outliers, as it may adversely impact the goal of maximizing the inter-class separation (including future classes, for new users). The final challenge is how to compose a new combined goal that prioritizes both consistency and distinction from the diverse goals of the two loss functions.

We first address the challenge of input mismatch of the two loss functions. Triplet loss selects triplets as input, but selecting tuples for triplets is difficult, and the performance and stability of the network depend on the correct order of the training set, which results in a weaker generalization capability. Instead of training the model as triplets, we train our model as mini-batch $\mathbb{D}_{MB} \subset \mathbb{D}$ in each iteration. We modify the formulation of Triplet Loss when composing the overall loss function.

We now define the specific distance metric we use in our loss function. As described earlier, Neural network $\phi : \mathbb{D} \rightarrow \mathcal{F}$ computes deep features for each sensor segment, where $\mathcal{F}$ is the feature space and $f_u^i = \phi(s_u^i)$ is the computed deep feature vector from sensor segment s_u^i. Let feature space be a metric space with L^2-norm.

Recall that $\mathcal{S}_u$ contains all the sensor segments of user u. The distance between sensor segment s_v^i and a class of sensor segments $\mathcal{S}_u$ is defined by the average distance between s_v^i and all other elements of $\mathcal{S}_u$ in the feature space,

$$d(s_v^i, \mathcal{S}_u) = \frac{1}{|\mathcal{S}_u|} \sum_{s_u^j \in \mathcal{S}_u} \|\phi(s_v^i) - \phi(s_u^j)\|_2^2.$$

We use this definition of distance metric instead of the distance from Centroid used in the Center Loss function in order to reduce the number of model parameters. We now describe the consistency and distinction metric before presenting our overall loss function.

Consistency (for Intra-class variation) of $\phi(s_u^i)$ is the average distance of point $\phi(s_u^i)$ from all other points $\phi(s_u^j)$ of same

class/user in feature space F. More formally, consistency of s_u^i is,

$$C(s_u^i) = d(s_u^i, \mathcal{S}_u)$$

Now, consistency of the Class u is defined as an aggregated function, ψ, of all the point consistencies in the class.

$$C_u = \psi\left(\{C(s_u^i)\}_{s_u^i \in \mathbb{D}_{MB}}\right)$$

We want this aggregated function to measure the sparsity of the class and not be susceptible to outliers (see the second challenge above). For this purpose, we can use a percentile measure for ψ. For our experiments, we use the 95^{th} percentile of the point consistency values of a class. Finally, consistency is defined by the mean consistency of all the classes.

$$C = \frac{\sum_{u \in \mathcal{I}_A} C_u}{n}$$

Distinction (for Inter-class variation) of $\phi(s_u^i)$ is the distance of point $\phi(s_u^i)$ from the closest point belonging to a different class in the feature space:

$$D(s_u^i) = \min_{v \in \mathcal{I}_A, v \neq u} d(s_u^i, \mathcal{S}_v)$$

Overall distinction is defined as the mean distinction of all points.

$$D = \frac{\sum_{s_u^i \in \mathbb{D}_{MB}} D(s_u^i)}{|\mathbb{D}_{MB}|}$$

To address the third challenge of composing an overall loss function that can concurrently optimize both consistency and distinction, we formulate our loss function using a similar formulation as triplet loss (by replacing positive and negative distances with consistency and distinction, respectively). More specifically, we propose our *Consistency-Distinction (CD)* Loss function as follows

$$\mathcal{L}_{CD} = C - D + \alpha * C$$

With this formulation, the deep learning model will minimize the value of loss function, resulting in minimizing consistency C and maximizing distinction D, until the value of D is at least $\alpha \times C$. Here, α is a threshold on the ratio that is enforced between intra-class distance and inter-class distance. We note that our formulation differs from the Triplet Loss that uses α as a constant threshold on the difference in pairwise distances. We instead apply α to the ratio between the intra-class distance and inter-class distance because our loss function is not measuring the distance between two points, but distance within and between two clusters.

For our proposed loss function to be acceptable in training of a deep learning model, we need to show that it is differentiable. We first note that since our distance function is a sum of several distances and each distance is differentiable; therefore, our distance function is differentiable. The gradient of $d(s_u^i, \mathcal{S}_v)$ with respect to a point in feature space f_u^i is,

$$\frac{\partial}{\partial f_u^i} d(s_u^i, \mathcal{S}_v) = \frac{1}{|\mathcal{S}_v|} \sum_{s_v^j \in \mathcal{S}_v} \left(\phi(s_u^i) - \phi(s_v^j)\right).$$

As our proposed loss function $\mathcal{L}_{CD}$ is a linear combination of multiple differentiable functions, our loss function is also differentiable. The gradient of $\mathcal{L}_{CD}$ with respect to f_u^i is computed as:

$$\frac{\partial \mathcal{L}_{CD}}{\partial f_u^i} = \frac{1}{m} \sum_{s_u^j \in \mathcal{S}_u} \left(\frac{\partial}{\partial f_u^i} d\left(s_u^j, \mathcal{S}_u\right) - \frac{\partial}{\partial f_u^i} d\left(s_u^j, \mathcal{S}_v\right)\right).$$

4.1.4 The Loss Function. We adopt the joint supervision of soft-max loss and CD loss to train our proposed neural network for discriminative feature learning. More specifically,

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda \mathcal{L}_{CD},$$

where $\mathcal{L}_{CE}$ is cross-entropy soft-max loss [79] and a scalar λ is used for balancing the two loss functions.

5 BOOSTING MODEL DESIGN

We first present two choices for the boosting model and then discuss considerations for selecting the unit length Δ for segmenting data.

5.1 Boosting Method

The boosting methods use the user ids produced by the base model on each data segment of a given sensor trace to improve the re-identification performance. For a given test data sample $\hat{s}_x$ of length l, we perform the following steps prior to boosting. We first partition $\hat{s}_x$ into $m = \left\lfloor \frac{l}{\Delta} \right\rfloor$ segments where each segment is of length Δ. Second, each segment is fed as input to the base model M_Δ, resulting in m likelihoods for each anonymized user $u \in \mathcal{I}_A$. We thus obtain a $m \times n$ matrix of likelihoods P. As shown in Figure 4, we consider two boosting methods: a) MaxMean and b) Majority, and compare their performance in experiments.

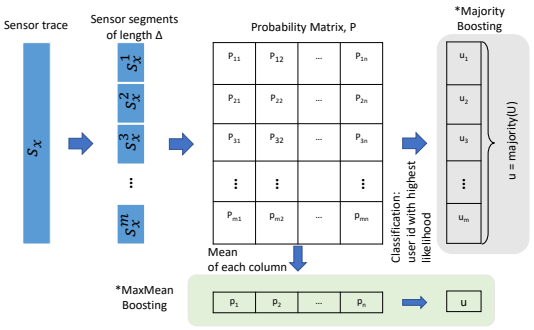


Figure 4: Overview of the boosting approaches.

MaxMean boosting: This boosting method creates a likelihood vector of size n from the likelihood matrix P by computing the mean likelihood for each user id. Finally, it outputs user id with the maximum likelihood, provided it is greater than the decision threshold $\mathcal{T}$, and outputs 0, otherwise.

Majority boosting: This boosting method replaces each row with the user id having the highest likelihood in that row (if it exceeds the decision threshold $\mathcal{T}$, and 0, otherwise). This step reduces the matrix of likelihoods to a vector with m most likely user id's. Finally, it reports the majority prediction from these m user id's. In both steps of majority assignment, any ties are broken randomly.

5.2 Selection of Segment Length Δ

The segment length (Δ) for accelerometry data is determined by the minimum amount of data that is sufficient to identify both distinction of activity pattern from other users and consistency with other segments from the same user. We note that the choice of Δ can have a substantial impact on re-identification performance.

First, we expect the performance of the base model to increase monotonically as we grow the value of Δ. This is because if a smaller value of Δ_1 has a better performance than a higher value of $\Delta_2 (> \Delta_1)$, then the base model can locate the segment of length Δ_1 within the large segment of Δ_2 to achieve at least the same performance as that when provided a sub-segment of length Δ_1. Intuitively, when the value of Δ is large (e.g., full day), it can capture different aspects of the user's motion patterns, revealing uniqueness and consistency in daily patterns such as routines. Therefore, a larger size of Δ increases the accuracy of the base model.

However, for a fixed length l of a test sample, as the value of Δ increases, the number of units that can be assessed by the base model decreases, reducing the opportunity to boost the re-identification performance by a boosting model. Hence, there is a trade-off between the value of Δ and the number of units of data assessed by the base model that can be used to boost the overall performance. As we show in experiments, the performance of the boosting model exhibits a convex function behavior, allowing us to select an appropriate value of Δ for a given test length l.

6 RE-IDENTIFICATION RISK ANALYSIS

6.1 Dataset

Our goal is to analyze re-identification risks from wrist-worn accelerometry data collected in the users' natural environment. For this purpose, we use a new dataset of raw accelerometery data from wrist-worn devices that resulted from the mPerf research study conducted to predict the work performance of employees using modeled data from wearable sensors. The study was approved by the Institutional Review Board (IRB Protocol # STUDY00000940 at the University of Minnesota-Twin Cities and an accompanying IRB Protocol # PRO-FY2018-161 at the University of Memphis). All participants provided written informed consent.

Each participant wore a wrist device (consisting of 3-axis accelerometers, 3-axis gyroscopes and a 3-channel Photoplethysmography sensor, all sampled at 25 Hz) and carried a smartphone with the mCerebrum study app [32] installed. They were asked to collect data for at least 8 hours each day for ten weeks (i.e., 70 days). The participants were knowledge-workers from diverse professions, including management, information technology, education, engineering, production, sales, transportation, etc., covering various job functions, ranging from senior executives to production personnel. A total of 380 participants completed all study procedures. Excluding participant-days when at least one hour of accelerometry data is not present (due to data loss, corruption, non-wear, or metadata mismatch), data from 353 participants (174 males, 123 females; mean age 31.7 ± 7.5 years) for a total of 190,078 hours of accelerometry data consisting of 51.3 billion data points were usable for this analysis. All data were stored and analyzed in Apache PySpark based Cerebral Cortex open-source platform [40].

6.2 Experiment Setup

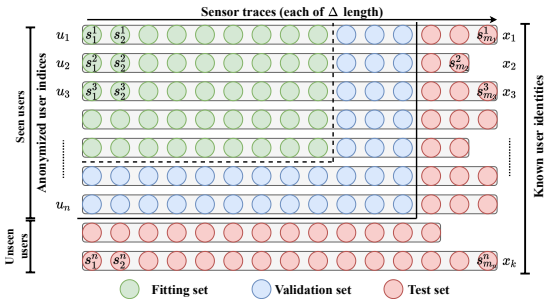


Figure 5: Splitting of training, validation, and testing set. The dataset is first divided into training and testing sets, and then the training set is further divided into a fitting set and a validation set containing a closed set and an open set.

Figure 5 describes how we organize our entire dataset for training, validation, and testing. For the training dataset, we randomly select 80% of the participants (282 out of 353) for training, leaving 20% (i.e., 71) participants to be only in the test set. Further, data from 80% of participants in the training set (i.e., 226) are used in the fitted set, leaving 56 participation for validation. Next, two-thirds of the sensor traces from all 282 training set participants are used to construct the training dataset (D_{train}). Within the fitted set of users, two-thirds of their training data segments are used in the fitted dataset. The remaining training data segments from these participants in the fitted set and two-thirds of all data from participants in the validation set are used for validation during training.

Test dataset for the adversary consists of one-third data from all 282 training participants and all data from 71 test participants. Since the adversary has access to some test data from each participant, $\mathcal{I}_K$ consists of known id of everyone. Anonymized set ($\mathcal{I}_A$) consists of anonymized index of 226 participants in the fitting set.

Since deep learning is compute-intensive, for the experiment on finding the best segment length, we use a smaller version of the validation dataset, namely $D_{100,50} \subset D_{train}$, with 100 participants in the fitted set and 50 participants in the validation set.

For data processing, we first segment the accelerometry data into one-minute windows. We retain a minute if it contains at least 85% of the expected number of samples. To take into account the effect of diversity in the available training data distribution, we conduct each experiment several times with different random seeds to select different windows of training data segments. For selecting a given length l of training data window for a participant, we randomly choose a starting point and select a contiguous segment of length l starting there. The test results are obtained by applying the trained models on the test data set aside from each participant. We conduct multiple iterations of the training window selection and report averages to obtain a robust measure of performance.

6.3 Performance Metrics

For our performance evaluation, we use True Matching Rate (TMR), and False Acceptance Rate (FAR), as defined in Section 2.2 (see (1)

and (2)). They are the two most commonly used evaluation metrics for open set recognition (OSR) [53]. For the evaluation of re-identification risk, we again use the definition from Section 2.2 (see (3)). We approximate $\rho_x = Pr[\exists u \in \mathcal{I}_A : \mathcal{K}(u) = x]$ in our experiments as the percentage of total participants whose data is used in the fitting set.

6.4 Model Architecture & Parameter Selection

In this section, we summarize the effects of varying the model hyperparameters on validation accuracy, compare the performance of alternative model architectures and demonstrate the effectiveness of the Consistency-Distinction (CD) loss function.

6.4.1 Selection of Δ. To determine the best segment length Δ, we evaluate different values (in seconds) from the set {5, 10, 20, 30, 45, 60, 90, 120}. We then analyze the performance of the boosting model for test samples of length (l) 5 minutes and 10 minutes.

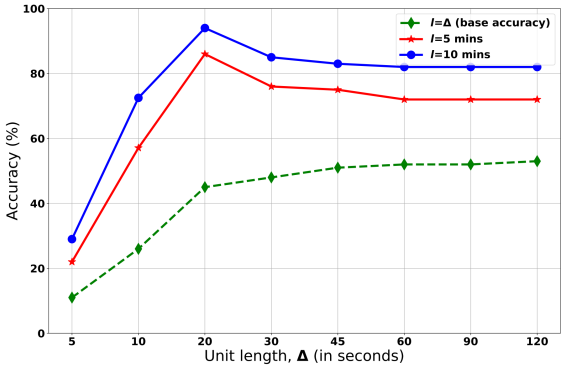
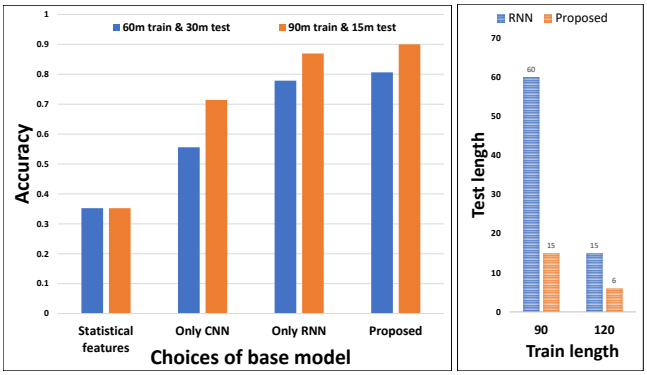


Figure 6: Validation accuracy of the boosting model for different values of Δ on the $D_{100,50}$ data set. The best accuracy occurs at $\Delta = 20$ seconds.

From Figure 6, we see that the performance of the boosting model approximates a convex function, peaking at $\Delta = 20$ seconds, also representing the best choice. We note that until $\Delta = 20$ seconds, the boosting model accelerates the performance gain from the base model. After $\Delta = 20$ seconds, the gain from the base model begins to saturate. Concurrently, the number of segments in the test set also reduces, resulting in a drop in performance at $\Delta = 30$ seconds, followed by gradual saturation.

6.4.2 Base Model Architectural Choices. Recall that our proposed base model consists of both convolutional and recurrent layers. In this experiment, we evaluate if a simpler model with a) convolutional layers only (CNN), or b) recurrent layers only (RNN), can provide comparable performance. The CNN model consists of two convolutional layers, two max-pooling layers, ReLU as an activation function, and a fully-connected layer at the end. For the RNN model, we use GRU units with a tanh activation function and 50% dropout. We also compare it with a shallow learning model. For the shallow model, we compute a set of features (as mentioned in [22]) from raw data and then employ a fully connected layer that takes this feature vector as input. The output of this layer is passed through a softmax layer.



(a) (b)

Figure 7: (a) Performance of our proposed model compared to Shallow Learning, CNN and RNN models. (b) The proposed model requires significantly lesser amount of test data to achieve the validation accuracy of 90% compared to the closest-performing RNN model.

Figure 7 (a) shows that the validation accuracy of our proposed model is higher than both the shallow and CNN models. Figure 7 (b) shows the test data lengths required by both the RNN and our proposed model for varying train lengths, and a fixed accuracy of 90%. While the accuracy of our proposed model (90%) is only slightly better than the RNN model (86%), it achieves the 90% accuracy with only 15 minutes of test data, but, for the RNN model, 60 minutes of test data is needed to achieve a similar accuracy.

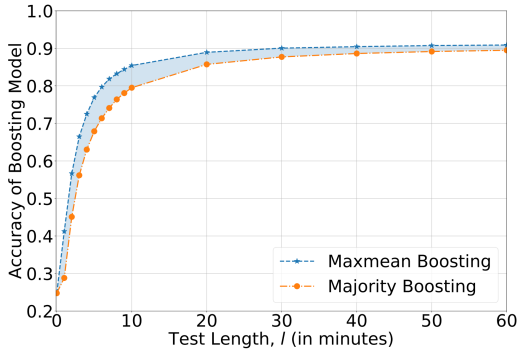


Figure 8: Performance of Majority and MaxMean boosting.

6.4.3 Choice of the Boosting Model. Figure 8 presents a comparison between Majority and MaxMean boosting. We observe that as the number of segments in the test sample increases (starting at Δ when no boosting occurs), MaxMean boosting performs better. A potential reason is that averaging across each iteration of the base model and then taking maximum provides greater robustness (in finding the best matching id) than picking an id with the maximum likelihood in each iteration and then selecting an id with the highest frequency. When the number of test unit segments are small, and ties are broken randomly, other id's with similar patterns may have

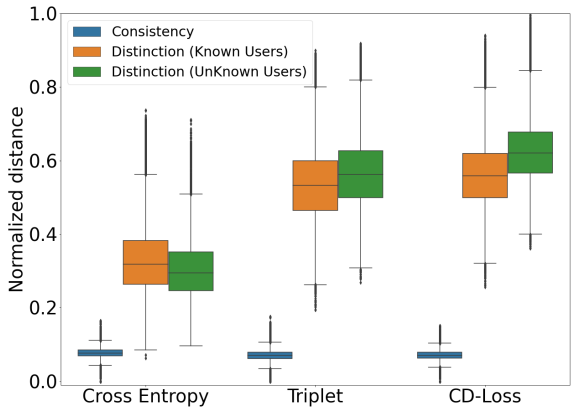


Figure 9: The distribution of consistency and distinction in terms of normalized distance from model trained on both with CD-Loss and without CD-Loss.

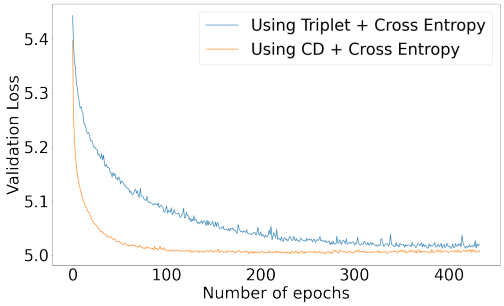


Figure 10: The model converges faster when using CD-loss due to its robustness in the presence of noisy data.

a fair chance of being selected. For larger test segments ($\geq$ 30 minutes), this difference in performance becomes negligible.

6.4.4 Choice of the Loss Function. To evaluate the impact of our Consistency-Distinction (CD) loss function on model performance, we determine the intra-class spread and inter-class distance (from the closest class), both in the same feature space. We normalize the set of both the distances to be in $[0, 1]$ for ease of visual comparison. The distribution of intra-class distances and inter-class distances from models trained with cross-entropy, Triplet, and CD-loss functions are shown in Figure 10. We observe that using our CD Loss function reduces the intra-class distance (improving consistency) and widens the inter-class distances, improving distinction. The CD Loss provides 10% improvement over Triplet Loss for distinction from unknown users (0.56 vs. 0.62). Qualitatively, CD loss improves negative mining challenge in Triplet Loss by selecting the closest negative instance vs. a random negative instance.

A second benefit of CD loss is its faster convergence. Presence of outliers that are prevalent in noisy data collected from the field environment, in the training data can slow down model convergence. Triplet loss is susceptible to such slowness due to its dependence

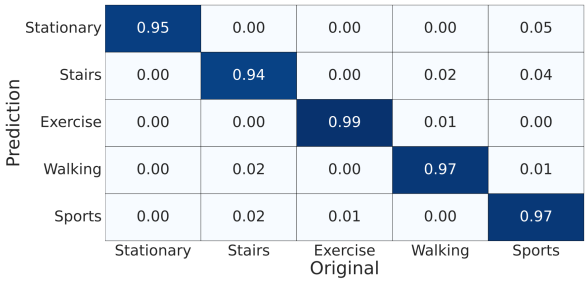


Figure 11: Confusion Matrix of Activity Classification in WISDM dataset

on the selection of a minibatch. By using aggregate (class) level approximation of consistency and distinction in the feature space, CD loss avoids such sensitivity on specific data points and hence it convergences faster in the presence of noisy data expected in real-life. With CD Loss, the model converges in 100 epochs vs. 400 when using Triplet Loss (see Figure 10).

6.5 Re-identification Risk Characterization

We now apply our model to characterize the re-identification risk when wrist-worn accelerometry data from daily life are shared. We first segment the day-long timeseries of data into broad classes of physical activity states. We only consider activities that can be detected from short 20-second data segments.

We train a Convolutional Neural Network (CNN) based activity recognition model for each 20-second data segment using publicly available WISDM dataset [72]. In WISDM, 51 participants performed 18 different activities while wearing accelerometers on their dominant wrists. Based on the amount of periodicity and variations present in different activity labels, we merge similar activities to obtain the following classes — *Stationary*, *Walking*, *Stairs*, *Sports*, and *Exercise*. *Stationary* refers to segments where the variation is minimum and encompass labels such as sitting, standing, typing and others. *Walking* incorporates activities when there is gait information present, with those involving *Stairs* separated out. *Sports* refers to activities which consist of a mixture of stationary and sudden burst of active segments. These include playing, catching, dribbling, etc. *Exercise* includes activities of high magnitude such as jogging, running and cycling. Although periodicity is observed in the data segments for both *Exercise* and *Walking*, the two are different based on the magnitude of variations present.

For generalizing across orientation differences in different devices and study setups, we train the model using only magnitude of accelerometer data. Using 20% of each participants data as testing set, our model achieves an accuracy of 0.96 and a weighted F1-score of 0.96. Figure 11 shows the confusion matrix. After the model is trained, we apply it on our dataset to obtain the activity labels.

The model's performance depends on the various properties of the database: the number of users in the database (n), the set of activities people perform in their daily life (A), the length of the sensor traces (L) used for training, and the length of the sensor

trace of the test sample (l). In this section, we analyze the effect of these parameters on the re-identification risk.

6.5.1 Impact of Activity Type on Re-identification Risk. In their daily life, users engage in a variety of different physical activities while wearing a wrist sensor such as a smartwatch. To analyze the impact of different activity states, we classify each data segment into one of the five activity classes of *Stationary, Walking, Stairs, Sports*, and *Exercise*. We then compare the re-identification risk associated with each of the different activities for the following combinations of train and test data lengths: a) train: 60 minutes, test: 30 minutes, b) train: 60 minutes, test: 60 minutes, c) train: 120 minutes, test: 30 minutes, and d) train: 120 minutes, test: 60 minutes. As shown in Figure 12, *Exercise* leads to the highest re-identification followed by *Walking*. For *Sports* activities, performance when trained with 60 minutes of data is low, but performance improves considerably with increase in training data length. Finally, data from the *Stationary* state poses the least re-identification risk.

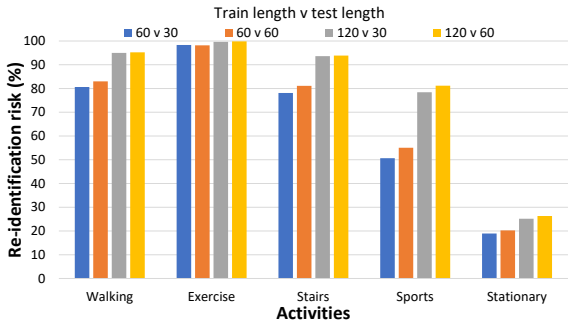


Figure 12: Activity-wise re-identification risk profile.

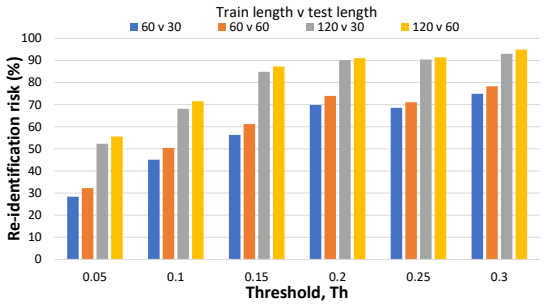


Figure 13: Re-identification risks for different standard deviation threshold values used to filter data.

6.5.2 Impact of Activity Intensity. To better understand why different activity types pose different re-identification risks, we analyze the impact of activity intensity on re-identification risk. We use a standard deviation (SD) threshold to indicate the intensity of physical activity as experienced by the wrist-worn accelerometers. Only segments with SD above a threshold (denoted by Th) are used to train and test the model. Figure 13 shows re-identification risks for different values of the threshold. As the threshold increases, we observe an improvement in re-identification accuracy. This observation indicates that higher intensity motion inherently carry higher re-identification fingerprints.

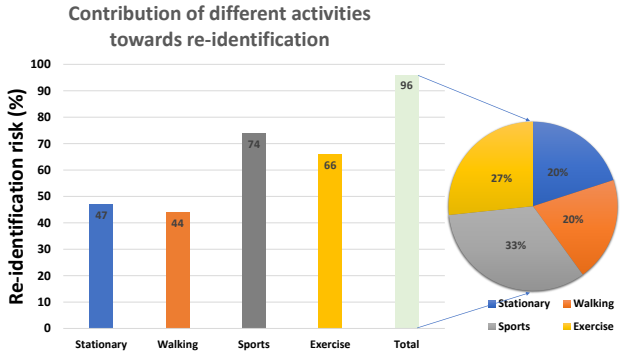
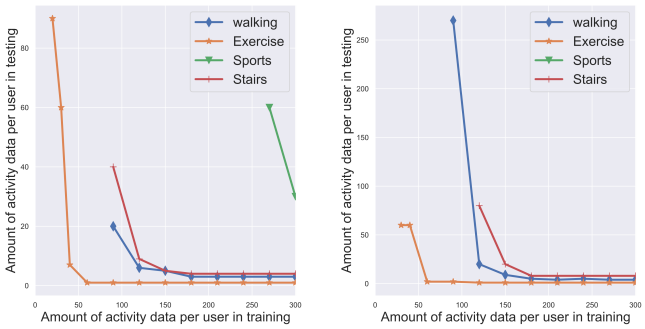


Figure 14: Activity-wise re-identification risk when user shares one day of wrist-worn accelerometry data.

6.5.3 Re-identification Risk from An Entire Day of Sensor Wearing. Section 6.5.1 presented the activity-wise re-identification risks if a user spends the same amount of time in each activity class, observing that *Exercise* carries the highest re-identification risk. But, users spend different amounts of time in each activity state. For example, in our dataset, users spend only 1% of their time in *Exercise* state. On the other hand, people remain mostly *Stationary* throughout the day (80% of the time as observed in our study). While the re-identification risk from stationary data is low, the large volume of data can still contain useful discriminatory patterns.

We observe that on average, people wear their device for about 10 hours a day. Out of which, they remain *Stationary* for about 8 hours. On an average, 30 minutes are spent on walking, and only 6 minutes in *Exercise*. The contribution of each of the above activities to re-identification risk is shown in Figure 14. To put these results in perspective, we further translate other activities' length in terms of the average length of the *Walking* activity. For example, the risk from 50 minutes of walking is 77%, which is the same as risk from about 90 minutes of *Non-stationary* activity; similarly, 40 minute of *Walking* has a similar re-identification risk as 6 minutes of *Exercise*, and finally, 30 minutes of *Walking* has similar re-identification risk as 480 minutes of remaining *Stationary*. Thus, the re-identification risk from 10 hours of sensor data is the same as the risk from 150 minutes of *Walking* data. Taken together, re-identification risk from one day of sensor data release is 96%.

6.5.4 Impact of Activity Duration on Re-identification Risk. To inform experiment designs that are conducted with an aim to publicly release the wrist-worn accelerometry datasets for research (e.g., to develop or validate activity recognition models), we analyze the time spent in different activity types that result in high re-identification risks. For a given risk level, η, and training length L, i.e., $|s_u| = L, \forall u \in \mathcal{I}_A$, we want to determine the minimum test length l, i.e., $|\hat{s}_x| = l$, such that $\mathcal{R}(\mathcal{D}) > \eta$. Consider the $2D$ plane, where x-axis represents the training data length, and y-axis represents the test data length. We partition the space into two subspaces



(a) When risk > 90% (b) When risk > 95%

Figure 15: Minimum amount of training and testing data to produce significant re-identification risks

such that for all the points of one subspace re-identification risks are lower than η and vice-versa. The re-identification risk monotonically increases if we fix either the train length or the test length and increase the other. Therefore, for each train length L, we find the minimum test length such that the risk > η. If we connect all the points, we get a separation line. Figure 15a presents such separation lines of each activity type for $\eta = 90\%$ and Figure 15b for $\eta = 95\%$. Detailed re-identification risk profiles for different activities over a grid of training and test lengths is visualized into multiple heat maps and presented in the Appendix (see Figures 19 and 20).

We observe that releasing even 40 minutes of exercise data can enable an adversary to train a re-identification model that can re-identify a user with only a few minutes of test data. But, for walking (a routine activity), one and half hour of data is needed to pose a high re-identification risk. We note that when a week or longer duration of data from daily life is released, it is likely to have adequate data for a significant re-identification risk.

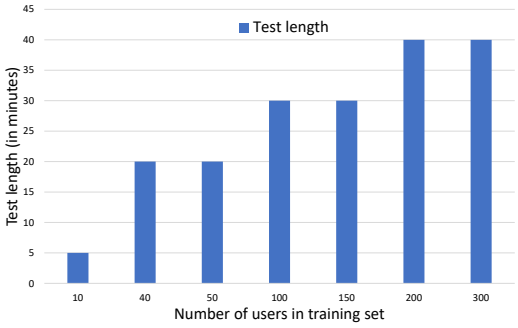


Figure 16: An increase in the number of fellow study participating reduces the re-identification risk. The bars indicate the corresponding test data lengths needed to achieve a re-identification accuracy of 80% (for *Sports* activities).

6.5.5 Impact of Number of Users (n) in the Study on Re-identification Risk. To understand the risk of participating in a large versus a small study, we analyze the impact of n on re-identification risk. Having more users in the same dataset makes it more likely to find users with similar fingerprints, reducing the re-identification risk. For this analysis, we select the *Sports* activity to explore a wider spectrum of re-identification risk. It has a lower re-identification risk when compared with *Walking*, *Stairs* or *Exercise*, but still has adequate re-identification risk unlike the *Stationary* state.

We fix the training data length of each user to 300 minutes, and plot the effect of the number of users in Figure 16. We observe that the trained model needs only five minutes of test data to achieve 80% re-identification accuracy for $n = 10$. But, as the population size increases, the amount of test data needed also increases to achieve the same level of re-identification accuracy.

6.6 Model Generalizability and Scalability

Practical limitations and associated costs of collecting large volumes of diverse training data often imply that systems end up overfitting to the limited available data. However, for our model-based re-identification system to be useful it needs to generalize and maintain a low FAR. Towards this end, we perform an entropy-based analysis to assess the scalability of our re-identification system.

Let n be the number of users, and d be the dimension of the latent representation (or feature space) used for performing re-identification. We use the output of the dense layer (output of FC2 layer in Figure 3) as the $d-$dimensional continuous feature subspace $F \subseteq \mathbb{R}^d$. We compute the differential entropy, $H_n(F)$, of the feature subspace, such that $2^{H_n(F)}$ roughly indicates the maximum volume of unique users that can be represented with no overlap, iff each user data had a unique support. However, due to natural variations in user activities, their data is often a localized distribution in the feature space. We use the per-user data distribution to compute the average differential entropy for a single user as $H_1(F)$. Thus, for a well-trained model, the differential entropy of the re-identification system is given by $H_n(F) - H_1(F)$. High system entropy value indicates lower chances of collision between the representation of any two users and better generalization capability.

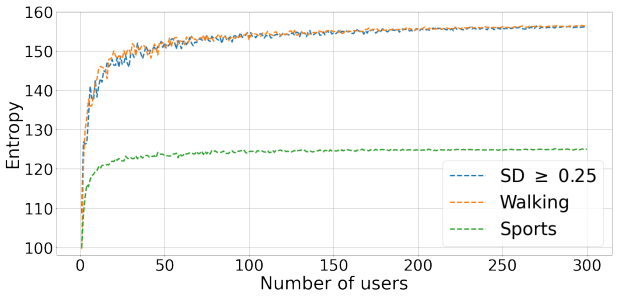


Figure 17: Change in system differential entropy as new user data are included for different activities. Saturation indicates that the feature space learned by the model is able to handle additional users without change.

6.6.1 Differential Entropy of the Re-identification System. To compute system differential entropy, we project all segments from all

the users into the feature space F. We then model the user data distribution in the feature space as a multivariate Gaussian $\mathcal{N}(\mu, \Sigma)$, where μ is the zero-mean vector, and the covariance matrix Σ is computed using the feature vectors. The differential entropy of a multivariate Gaussian is given by, $H(F) = \frac{1}{2}ln(|\Sigma|) + \frac{d}{2}(1 + ln(2\pi))$ where $|\Sigma|$ is the determinant of the covariance matrix.

We use the data from 353 users to compute $H_{353}(F)$. Figure 17, shows the change in differential entropy of the system $(H_{353}(F) - H)_1(F))$ as new users are included in the system for different activities (*Sports*, *Walking*) and SD threshold based activity classification. We observe that the plots, for each activity, start to saturate at some point indicating that the feature space does not change appreciably as more users are added. This leads us to believe that the model has generalized well to the population represented by the training dataset. We also compute the system differential entropy for each activity and find that for *Walking* that most users routinely engage in, the differential entropy is 56 bits, translating to $\approx 7.2x10^{16}$ users.

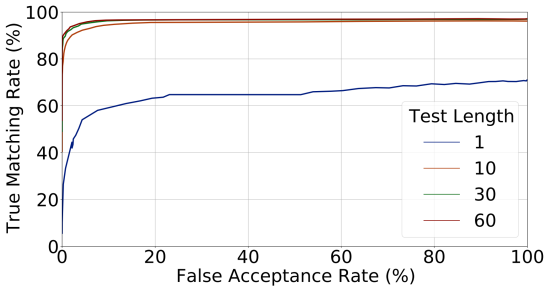


Figure 18: ROC for Different Choices of Test Data Length

6.6.2 Evaluating Representational Overlap. Overlap in user representation in the feature space adversely impacts both TMR and FAR. Figure 18 shows an ROC curve of the boosting model for different lengths of test data (l) for the activity state of *Walking*. For $l = 10$ minutes, the model achieves a 90.25% TMR and 2.16% FAR. As we increase l to 30 minutes, the model achieves more than 94.06% TMR for an FAR of 3.49%. If the model is provided with test data of 60 minutes, the model achieves a TMR of 94% while keeping the FAR to 1.75%. This shows that users are represented with minimal overlap allowing for their accurate re-identification.

6.6.3 False Acceptance Rate for An Independent Dataset. To further test the utility of our open set formulation and new CD loss function in achieving generalizability of the presented model, we compute the FAR for a publicly available mORAL dataset [4]. In this dataset, 25 participants collected wrist-worn accelerometer and gyroscope data throughout the day for one week continuously. The brushing and flossing events are labeled from self-recorded videos. We apply the trained activity detection model on this data to identify the segments belonging to different activity types. We then select an operating point for our model on the ROC curve corresponding to different TMR and FAR values. Using these optimized decision thresholds, we compute the false acceptance rate for the mORAL participants. We obtain FAR values of 2.26% (TMR $\geq$ 90%), 2.04% (TMR $\geq$ 94%) and 1.02% (TMR $\geq$ 94%) in mORAL data for test lengths of 10, 30 and 60 minutes, respectively. This is similar to what we observe in Section 6.6.2 when FAR and TMR values are calculated from the original dataset. These results further confirm the utility of our open set formulation and our CD loss function.

7 RELATED WORKS

There is a growing body of work on discovering and mitigating security and privacy problems in human-cyber-physical systems that emanate from continuous collection of sensor data from wearable devices carried by users in their daily life [2, 9, 11, 12, 14, 49, 63, 65, 66, 78]. They use methods drawn from signal processing, information theory, and machine learning. In the following, we focus on works that are closely related to user re-identification.

Sensor data-based approaches for re-identification can be grouped into two categories — behavioral biometric approaches and device fingerprinting. Behavioral biometric approaches have been used for user authentication in several research works. Examples include [76] and [64] that use hand waving detected from two different sources (accelerometer and ambient light sensor) to authenticate a user in smartphones. Others [20, 24, 25, 34, 39, 61] feed keystroke biometrics and touchscreen interaction pattern (key pressed location, duration of keypress, size, drift, etc.) in different machine learning models to authenticate phone users. These methods are not directly applicable to person re-identification from wrist-worn accelerometry data because they rely on scripted settings (e.g., waving a hand or holding the phone in hand).

Several LSTM based user authentication methods, i.e., DeepAuth [7, 66] and AUToSen [1], use accelerometer and gyroscope data from the smartphone to capture behavioral patterns with high accuracy. A learning based method called RiskCog [80] validates users using data collected from accelerometer, gyroscope, and gravity sensors with high accuracy. We show that our model trained with the proposed CD loss function for an open set formulation outperforms these models in the amount of training data needed.

Another popular approach to behavioral biometric is gait-based authentication [22, 26, 41, 46, 55]. These approaches extract gait-based unique fingerprint from physical activities such as walking or running, using motion data from accelerometers placed on different body locations, sometimes supplemented with a video. Recent research on gait based person identification uses a variation of Deep Neural Networks to achieve high accuracy [1, 77], establishing the feasibility of extracting unique characteristics of the user from their motion pattern. But, the applicability of these methods is limited due to their reliance on multiple sensors placed on different locations of the human body. Also, their methods are trained to learn a similarity function that measures matching scores of two templates given the condition that the users performed a specific activity, which is unlikely when users live freely in their natural environment. Therefore, none of the existing behavioral biometric solutions show the feasibility of person re-identification from wrist-worn accelerometry data collected from the natural environment.

Another complementary body of work seeks to re-identify a device (and subsequently a user, if the device is not shared among multiple users and until the user changes the device, e.g., upgrades their phone). These works, referred to as device fingerprinting, aim to generate a unique signature, or fingerprint, that uniquely

recognizes a specific device. Several works find the fingerprint by extracting statistical features and using supervised machine learning approaches when the phone vibrates (for example, during an incoming call or message) [18] or when stationary [15, 16]. These methods were found to have an F1 score of 60% in field setting when devices are held in hand.

Bojinov et. al. [8] models the imprecision in accelerometer calibration via a device-specific scaling and translation of the measured values. For analysis, they collected data when the device was stationary, achieving a re-identification rate of 53% for devices in their dataset. More recently, [78] estimated the calibration matrix more accurately by considering all three errors: scaling factor, bias, and non-orthogonality misalignment errors. All of these methods model the error of the sensor due to the hardware imperfections during the sensor manufacturing process. Our work is complementary to these works as we seek to extract distinctive and unique features from the patterns of micro-movements of a user's wrist.

Finally, privacy research on leakage of training data with the release of trained models investigate membership inference attacks [30, 50, 62, 69] to determine whether a specific data point belonged to the training set. Their focus is to find an exact match of a test sample with one in the training set, by exploiting the higher prediction confidence that models usually report when tested on their training data. Similar to our base model, they also use the likelihood produced by the model. But, these methods do not address our problem of data segmentation, construction of base model architecture that extracts the unique common micro-movement pattern for each class (i.e., person), or discovery of a loss function for the base model to minimize the intra-class distance in the feature space and simultaneously maximize inter-class separation, which are technical contributions of our work.

8 LIMITATIONS AND FUTURE WORKS

Although our *WristPrint* method achieves a 96% re-identification rate, there are several limitations to the presented work that open up numerous opportunities for future research.

First, in our dataset, each user's data came from the same device. Different wrist-worn devices differ in sampling rates, sensitivity range, mounting orientations, etc. This work did not experiment with these variations, and hence their impact on re-identification performance can be investigated in future works. More specifically, a higher sampling rate and lower noise of the signal may allow the model to capture finer-grained micro-movements, potentially improving re-identification performance and reducing the amount of data needed for training and testing for a specified level of performance. Future work can also investigate the case when the model is trained on data from one device but tested on another device.

Second, for this analysis, we only looked at the wrist-worn device. Motion sensors are included in wearable devices such as earbuds and smart eyeglasses that are worn on different body locations. Future work can investigate the suitability of the presented modeling approach for re-identification using motion data from such devices.

Third, our experiments show that the distinctive features of the user's wrist movement remain consistent for ten weeks. Future work can investigate the deterioration in re-identification performance over time as user's movement patterns evolve, especially after major events such as accidents, pregnancy, and job changes.

Fourth, our experiments show that the impact of segmentation length choice (Δ) on re-identification performance exhibits a convex shape, displaying unique optimal value for a given test data length (see Figure 6). Future work can develop theoretical frameworks to prove such a property and derive optimal values of Δ analytically.

Finally, future work can investigate how the re-identification risk increases when other sensing modalities included in wrist-worn devices (e.g., gyroscopes and pulse plethysmograph (PPG) for heart rate measurements) are used together with accelerometry data. Using additional sensing modalities can potentially reduce the amount of data needed for training and testing.

9 CONCLUSIONS

Several modalities of data are routinely used for user re-identification and sometimes even for authentication. They include video, voice, and fingerprints. But, new modalities of data are emerging that capture users' movement patterns at a very fine granularity. Wrist-worn devices have emerged as one such increasingly popular device. To support research for new inferences of daily behaviors from these devices, data collected from user studies are publicly shared, assuming a lack of any identifying information embedded in them. Our work shows that data collected from such devices, even at 25 Hz, can support user re-identification with 96% accuracy. This creates new research opportunities to address the new privacy, security, and ethical challenges.

10 ACKNOWLEDGEMENTS

This research was supported in part by the National Institutes of Health (NIH) under award P41EB028242, by the National Science Foundation (NSF) under awards ACI-1640813, CNS-1823221, CNS-1705135, and CNS-1822935, and by the Combat Capabilities Development Command (DEVCOM), Army Research Laboratory (ARL) under Cooperative Agreement Number W911NF-13-2-0045 (ARL Cyber Security CRA). The mPerf research study was supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA Contract No. 2017-17042800006. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of ODNI, IARPA, DEVCOM, ARL, NSF, NIH, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes not withstanding any copyright notation therein.

The authors also wish to thank Shahin Samiei, Dr. Timothy Hnat, and Dr. Syed Monowar Hossain from MD2K Center of Excellence at University of Memphis and Dr. Stephan Dilchert, Dr. Kevin Stanek, Dr. Brittany Mercado, and Dr. Adib Birkland, for their contributions to data collection and/or software used for data collection. The authors also thank Dr. Nirman Kumar from University of Memphis for brainstorming and careful critique.

REFERENCES

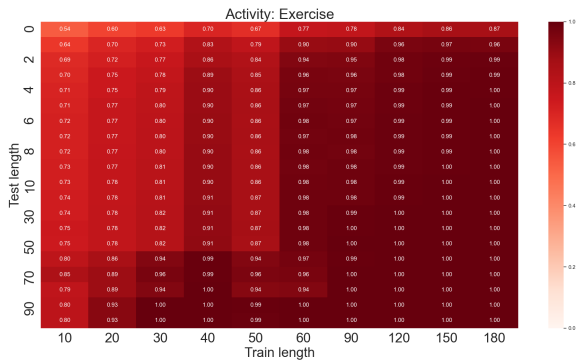
[1] Mohammed Abuhamad, Tamer Abuhmed, David Mohaisen, and DaeHun Nyang. 2020. AUToSen: Deep Learning-based Implicit Continuous Authentication Using

Smartphone Sensors. *IEEE Internet of Things Journal* 7, 6 (2020), 5008–5020.
[2] Shashank Agrawal, Saikrishna Badrinarayanan, Pratyay Mukherjee, and Peter Rindal. 2020. Game-Set-MATCH: Using Mobile Devices for Seamless External-Facing Biometric Matching. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 1351–1370.
[3] Sayma Akther, Nazir Saleheen, Mithun Saha, Vivek Shetty, and Santosh Kumar. 2021. mTeeth: Identifying Brushing Teeth Surfaces Using Wrist-worn Inertial Sensors. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)* 5, 2 (2021), 1–25.
[4] Sayma Akther, Nazir Saleheen, Shahin Alan Samiei, Vivek Shetty, Emre Ertin, and Santosh Kumar. 2019. mORAL: An mHealth Model for Inferring Oral Hygiene Behaviors in-the-wild Using Wrist-worn Inertial Sensors. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)* 3, 1 (2019), 1.
[5] Amin Ahsan Ali, Syed Monowar Hossain, Karen Hovsepian, Md Mahbubur Rahman, Kurt Plarre, and Santosh Kumar. 2012. mPuff: Automated Detection of Cigarette Smoking Puffs from Respiration Measurements. In *Proceedings of the International Symposium on Information Processing in Sensor Networks (IPSN)*. ACM, 269–280.
[6] Oliver Amft, Holger Junker, and Gerhard Troster. 2005. Detection of Eating and Drinking Arm Gestures Using Inertial Body-worn Sensors. In *Proceedings of the IEEE International Symposium on Wearable Computers (ISWC)*. 160–163.
[7] Sara Amini, Vahid Noroozi, Amit Pande, Satyajit Gupte, Philip S Yu, and Chris Kanich. 2018. Deepauth: A Framework for Continuous User Re-authentication in Mobile Apps. In *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*. 2027–2035.
[8] Hristo Bojinov, Yan Michalevsky, Gabi Nakibly, and Dan Boneh. 2014. Mobile Device Identification via Sensor Fingerprinting. *arXiv preprint arXiv:1408.1416* (2014).
[9] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z Morley Mao. 2019. Adversarial Sensor Attack on LIDAR-based Perception in Autonomous Driving. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2267–2281.
[10] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. 2017. Beyond Triplet Loss: A Deep Quadruplet Network for Person Re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 403–412.
[11] Yushi Cheng, Xiaoyu Ji, Juchuan Zhang, Wenyuan Xu, and Yi-Chao Chen. 2019. DeMiCPU: Device Fingerprinting with Magnetic Signals Radiated by CPU. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 1149–1170.
[12] Eunyong Cheon, Yonghwan Shin, Jun Ho Huh, Hyoungshick Kim, and Ian Oakley. 2020. Gesture Authentication for Smartphones: Evaluation of Gesture Password Selection Policies. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*. IEEE, 249–267.
[13] SH Cho, JM Park, and OY Kwon. 2004. Gender Differences in Three Dimensional Gait Analysis Data from 98 Healthy Korean Adults. *Clinical Biomechanics* 19, 2 (2004), 145–152.
[14] Anupam Das, Gunes Acar, Nikita Borisov, and Amogh Pradeep. 2018. The Web's Sixth Sense: A Study of Scripts Accessing Smartphone Sensors. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 1515–1532.
[15] Anupam Das, Nikita Borisov, and Matthew Caesar. 2016. Tracking Mobile Web Users Through Motion Sensors: Attacks and Defenses.. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*.
[16] Anupam Das, Nikita Borisov, and Edward Chou. 2018. Every Move You Make: Exploring Practical Issues in Smartphone Motion Sensor Fingerprinting and Countermeasures. *Proceedings on Privacy Enhancing Technologies* 2018, 1 (2018), 88–108.
[17] Erhan Davarci, Betul Soysal, Imran Erguler, Sabri Orhun Aydin, Onur Dincer, and Emin Anarim. 2017. Age Group Detection Using Smartphone Motion Sensors. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*. IEEE, 2201–2205.
[18] Sanorita Dey, Nirupam Roy, Wenyuan Xu, Romit Roy Choudhury, and Srihari Nelakuditi. 2014. AccelPrint: Imperfections of Accelerometers Make Smartphones Trackable.. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*.
[19] Dua Dheeru and Efi Karra Taniskidou. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml
[20] Benjamin Draffin, Jiang Zhu, and Joy Zhang. 2013. Keysens: Passive User Authentication Through Micro-behavior Mmodeling of Soft Keyboard Interaction. In *Proceedings of the International Conference on Mobile Computing, Applications, and Services (MobiSys)*. Springer, 184–201.
[21] Cynthia Dwork. 2008. Differential Privacy: A Survey of Results. In *Proceedings of the International Conference on Theory and Applications of Models of Computation*. Springer, 1–19.
[22] Muhammad Ehatisham-ul Haq, Muhammad Awais Azam, Jonathan Loo, Kai Shuang, Syed Islam, Usman Naeem, and Yasar Amin. 2017. Authentication of Smartphone Users Based on Activity Recognition and Mobile Sensing. *Sensors* 17, 9 (2017), 2043.
[23] Tim Grance Erika McCallister and Karen Scarfone. 2010. Guide to Protecting the Confidentiality of Personally Identifiable Information (PII). https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-122.pdf.
[24] Tao Feng, Xi Zhao, Bogdan Carbunar, and Weidong Shi. 2013. Continuous Mobile Authentication Using Virtual Key Typing Biometrics. In *Proceedings of the IEEE International Conference on Trust, Security and Privacy in Computing and Communications*. IEEE, 1547–1552.
[25] Mario Frank, Ralf Biedert, Eugene Ma, Ivan Martinovic, and Dawn Song. 2012. Touchalytics: On the Applicability of Touchscreen Input as A Behavioral Biometric for Continuous Authentication. *IEEE Transactions on Information Forensics and Security* 8, 1 (2012), 136–148.
[26] Davrondzhon Gafurov, Kirsi Helkala, and Torkjel Søndrol. 2006. Biometric Gait Authentication Using Accelerometer Sensor. *JCP* 1, 7 (2006), 51–59.
[27] Simson L. Garfinkel. 2015. De-identification of Personal Information. https://nvlpubs.nist.gov/nistpubs/ir/2015/nist.ir.8053.pdf.
[28] GDPR. 2021. Data Anonymization and GDPR Compliance: The Case of Taxa 4×35. https://gdpr.eu/data-anonymization-taxa-4x35/. [Online; accessed 06-Sept-2021].
[29] Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. 2020. Recent Advances in Open Set Recognition: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).
[30] Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. 2019. LOGAN: Membership Inference Attacks Against Generative Models. *Proceedings on Privacy Enhancing Technologies* 2019, 1 (2019), 133–152.
[31] Daojing He, Sammy Chan, and Mohsen Guizani. 2015. User Privacy and Data Trustworthiness in Mobile Crowd Sensing. *IEEE Wireless Communications* 22, 1 (2015), 28–34.
[32] Syed Monowar Hossain, Timothy Hnat, Nazir Saleheen, Nusrat Jahan Nasrin, Joseph Noor, Bo-Jhang Ho, Tyson Condie, Mani Srivastava, and Santosh Kumar. 2017. mCerebrum: A Mobile Sensing Software Platform for Development and Validation of Digital Biomarkers and Interventions. In *Proceedings of the ACM Conference on Embedded Network Sensor Systems (SenSys)*. 1–14.
[33] Jingyu Hua, Zhenyu Shen, and Sheng Zhong. 2016. We Can Track You If You Take the Metro: Tracking Metro Riders Using Accelerometers on Smartphones. *IEEE Transactions on Information Forensics and Security* 12, 2 (2016), 286–297.
[34] Elliu Huang, Fabio Di Troia, Mark Stamp, and Preethi Sundaravaradhan. 2021. A New Dataset for Smartphone Gesture-based Authentication. (2021).
[35] Ankita Jain and Vivek Kanhangad. 2016. Investigating Gender Recognition in Smartphones Using Accelerometer and Gyroscope Sensor Readings. In *2016 international conference on computational techniques in information and communication technologies (ICCTICT)*. IEEE, 597–602.
[36] Anil Jain, Arun Ross, and Salil Prabhakar. 2001. Fingerprint Matching Using Minutiae and Texture Features. In *Proceedings International Conference on Image Processing (Cat. No. 01CH37205)*, Vol. 3. IEEE, 282–285.
[37] Anil K Jain, Salil Prabhakar, Lin Hong, and Sharath Pankanti. 2000. Filterbank-based Fingerprint Matching. *IEEE Transactions on Image Processing* 9, 5 (2000), 846–859.
[38] Bargav Jayaraman and David Evans. 2019. Evaluating Differentially Private Machine Learning in Practice. In *Proceedings of the USENIX Security Symposium (USENIX Security 19)*. 1895–1912.
[39] Georgios Kambourakis, Dimitrios Damopoulos, Dimitrios Papamartzivanos, and Emmanouil Pavlidakis. 2016. Introducing Touchstroke: Keystroke-based Authentication System for Smartphones. *Security and Communication Networks* 9, 6 (2016), 542–554.
[40] Santosh Kumar, Gregory Abowd, William T Abraham, Mustafa Al'Absi, Duen Horng Chau, Emre Ertin, Deborah Estrin, Deepak Ganesan, Timothy Hnat, Syed Monowar Hossain, et al. 2017. Center of Excellence for Mobile Sensor Data-to-Knowledge (MD2K). *IEEE pervasive computing* 16, 2 (2017), 18–22.
[41] Jennifer R Kwapisz, Gary M Weiss, and Samuel A Moore. 2010. Cell Phone-based Biometric Identification. In *Proceedings of the IEEE International Conference on Biometrics: Theory Applications and Systems (BTAS)*. IEEE, 1–7.
[42] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. 2007. T-closeness: Privacy Beyond k-Anonymity and l-Diversity. In *Proceedings of the IEEE International Conference on Data Engineering (ICDE)*. IEEE, 106–115.
[43] Changchang Liu, Supriyo Chakraborty, and Prateek Mittal. 2017. DEEProtect: Enabling Inference-based Access Control on Mobile Sensing Applications. *CoRR* abs/1702.06159 (2017).
[44] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. 2007. l-Diversity: Privacy Beyond k-Anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 1, 1 (2007), 3–es.
[45] Mohammad Malekzadeh, Richard G Clegg, Andrea Cavallaro, and Hamed Haddadi. 2018. Protecting Sensory Data Against Sensitive Inferences. In *Proceedings of the Workshop on Privacy by Design in Distributed Systems*. ACM, 2.
[46] Jani Mantyjarvi, Mikko Lindholm, Elena Vildjiounaite, S-M Makela, and HA Ailisto. 2005. Identifying users of portable devices from gait pattern with accelerometers. In *Proceedings of the IEEE International Conference on Acoustics,*

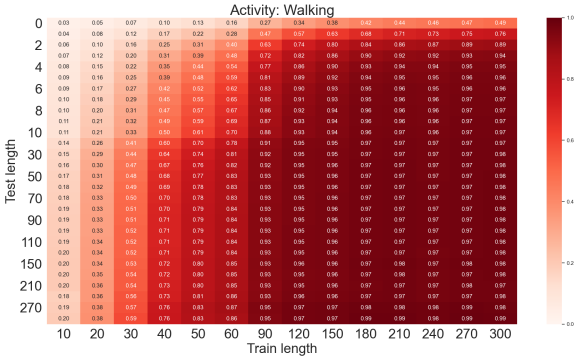
Speech, and Signal Processing (ICASSP), Vol. 2. IEEE, ii–973.
[47] Stephen M Mattingly, Julie M Gregg, Pino Audia, Ayse Elvan Bayraktaroglu, Andrew T Campbell, Nitesh V Chawla, Vedant Das Swain, Munmun De Choudhury, Sidney K D'Mello, Anind K Dey, et al. 2019. The Tesserae Project: Large-scale, Longitudinal, in Situ, Multimodal Sensing of Information Workers. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*. 1–8.
[48] David R Myers, Alexander Weiss, Margo R Rollins, and Wilbur A Lam. 2017. Towards Remote Assessment and Screening of Acute Abdominal Pain Using Only A Smartphone with Native Accelerometers. *Scientific reports* 7, 1 (2017), 1–12.
[49] Sashank Narain, Triet D Vo-Huu, Kenneth Block, and Guevara Noubir. 2016. Inferring User Routes and Locations Using Zero-permission Mobile Sensors. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*. IEEE, 397–413.
[50] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2018. Machine Learning with Membership Privacy Using Adversarial Regularization. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 634–646.
[51] Abhinav Parate, Meng-Chieh Chiu, Chaniel Chadowitz, Deepak Ganesan, and Evangelos Kalogerakis. 2014. RisQ: Recognizing Smoking Gestures with Inertial Sensors on A Wristband. In *Proceedings of the Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*. ACM, 149–161.
[52] Neha Patki, Roy Wedge, and Kalyan Veeramachaneni. 2016. The Synthetic Data Vault. In *Proceedings of the International Conference on Data Science and Advanced Analytics (DSAA)*. 399–410.
[53] P Jonathon Phillips, Patrick Grother, and Ross Micheals. 2011. Evaluation Methods in Face Recognition. In *Handbook of face recognition*. Springer, 551–574.
[54] Benny Pinkas. 2002. Cryptographic Techniques for Privacy-Preserving Data Mining. *SIGKDD Explor. Newsl.* 4, 2 (Dec. 2002), 12–19.
[55] Abena Primo, Vir V Phoha, Rajesh Kumar, and Abdul Serwadda. 2014. Context-aware Active Authentication Using Smartphone Accelerometer Measurements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. 98–105.
[56] RAAMP2. 2020. Rapid Automatic & Adaptive Model for Performance Prediction (RAAMP2) Dataset.
[57] George Rigas, Alexandros T Tzallas, Markos G Tsipouras, Panagiota Bougia, Evanthia E Tripoliti, Dina Baga, Dimitrios I Fotiadis, Sofia G Tsouli, and Spyridon Konitsiotis. 2012. Assessment of Tremor Activity in the Parkinson's Disease Using A Set of Wearable Sensors. *IEEE Transactions on Information Technology in Biomedicine* 16, 3 (2012), 478–487.
[58] Nazir Saleheen, Amin Ahsan Ali, Syed Monowar Hossain, Hillol Sarker, Soujanya Chatterjee, Benjamin Marlin, Emre Ertin, Mustafa al'Absi, and Santosh Kumar. 2015. puffMarker: A Multi-sensor Approach for Pinpointing the Timing of First Lapse in Smoking Cessation. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*. ACM, 999–1010.
[59] Nazir Saleheen, Supriyo Chakraborty, Nasir Ali, Md Mahbubur Rahman, Syed Monowar Hossain, Rummana Bari, Eugene Buder, Mani Srivastava, and Santosh Kumar. 2016. mSieve: Differential Behavioral Privacy in Time Series of Mobile Sensor Data. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*. 706–717.
[60] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A Unified Embedding for Face Recognition and Clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 815–823.
[61] Muhammad Shahzad, Alex X Liu, and Arjmand Samuel. 2013. Secure Unlocking of Mobile Touch Screen Devices by Simple Gestures: You Can See It But You Can Not Do It. In *Proceedings of the Annual International Conference on Mobile Computing & Networking (MobiCom)*. 39–50.
[62] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership Inference Attacks Against Machine Learning Models. In *IEEE Symposium on Security and Privacy (S&P)*. IEEE, 3–18.
[63] Yasser Shoukry, Paul Martin, Yair Yona, Suhas Diggavi, and Mani Srivastava. 2015. Pycra: Physical Challenge-response Authentication for Active Sensors Under Spoofing Attacks. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 1004–1015.
[64] Babins Shrestha, Nitesh Saxena, and Justin Harrison. 2013. Wave-to-Access: Protecting Sensitive Mobile Device Services via A Hand Waving Gesture. In *International Conference on Cryptology and Network Security*. Springer, 199–217.
[65] Akash Deep Singh, Luis Garcia, Joseph Noor, and Mani Srivastava. 2021. I Always Feel Like Somebody's Sensing Me! A Framework to Detect, Identify, and Localize Clandestine Wireless Sensors. In *Proceedings of the {USENIX} Security Symposium ({USENIX} Security)*.
[66] Payap Sirinam, Mohsen Imani, Marc Juarez, and Matthew Wright. 2018. Deep Fingerprinting: Undermining Website Fingerprinting Defenses with Deep Learning. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 1928–1943.
[67] Latanya Sweeney. 2002. k-Anonymity: A Model for Protecting Privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10, 05 (2002), 557–570.
[68] Edison Thomaz, Irfan Essa, and Gregory D Abowd. 2015. A Practical Approach for Recognizing Eating Moments with Wrist-mounted Inertial Sensing. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*. ACM, 1029–1040.
[69] Stacey Truex, Ling Liu, Mehmet Emre Gursoy, Lei Yu, and Wenqi Wei. 2018. Towards Demystifying Membership Inference Attacks. *arXiv preprint arXiv:1807.09173* (2018).
[70] Yonatan Vaizman, Katherine Ellis, and Gert Lanckriet. 2017. Recognizing Detailed Human Context in The Wild from Smartphones and Smartwatches. *IEEE Pervasive Computing* 16, 4 (2017), 62–74.
[71] Ziqi Wang, Brian Wang, and Mani Srivastava. 2021. Protecting User Data Privacy with Adversarial Perturbations. In *Proceedings of the 20th International Conference on Information Processing in Sensor Networks (co-located with CPS-IoT Week 2021)*. 386–387.
[72] Gary M Weiss. 2019. WISDM Smartphone and Smartwatch Activity and Biometrics Dataset. *UCI Machine Learning Repository, WISDM Smartphone and Smartwatch Activity and Biometrics Dataset Data Set* (2019).
[73] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. 2016. A discriminative feature learning approach for deep face recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 499–515.
[74] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. 2019. Modeling Tabular data using Conditional GAN. In *Advances in Neural Information Processing Systems (NeurIPS)*.
[75] Hiro-Fumi Yanai and Atsushi Enjyoji. 2016. Estimating Carrier's Height by Accelerometer Signals of a Smartphone. In *International Conference on Human-Computer Interaction*. Springer, 542–546.
[76] Lei Yang, Yi Guo, Xuan Ding, Jinsong Han, Yunhao Liu, Cheng Wang, and Changwei Hu. 2014. Unlocking Smart Phone Through Handwaving Biometrics. *IEEE Transactions on Mobile Computing* 14, 5 (2014), 1044–1055.
[77] Cheng Zhang, Wu Liu, Huadong Ma, and Huiyuan Fu. 2016. Siamese Neural Network Based Gait Recognition for Human Identification. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2832–2836.
[78] Jiexin Zhang, Alastair R Beresford, and Ian Sheret. 2019. Sensorid: Sensor Calibration Fingerprinting for Smartphones. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*. IEEE, 638–655.
[79] Zhilu Zhang and Mert Sabuncu. 2018. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. In *Advances in Neural Information Processing Systems (NeurIPS)*. 8778–8788.
[80] Tiantian Zhu, Zhengyang Qu, Haitao Xu, Jingsi Zhang, Zhengyue Shao, Yan Chen, Sandeep Prabhakar, and Jianfeng Yang. 2019. RiskCog: Unobtrusive Real-time User Authentication on Mobile Devices in the Wild. *IEEE Transactions on Mobile Computing* 19, 2 (2019), 466–483.

A APPENDIX

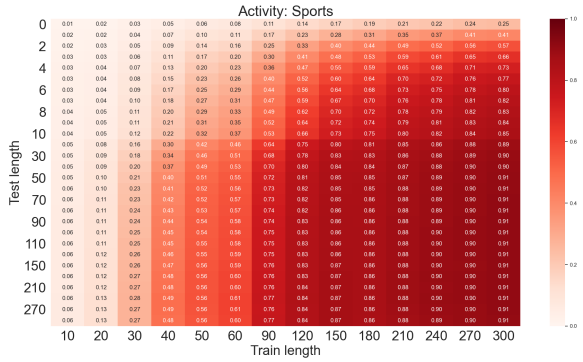
See Section 6.5.4 for a description of the figures presented here as well as for the notations used in the figures.



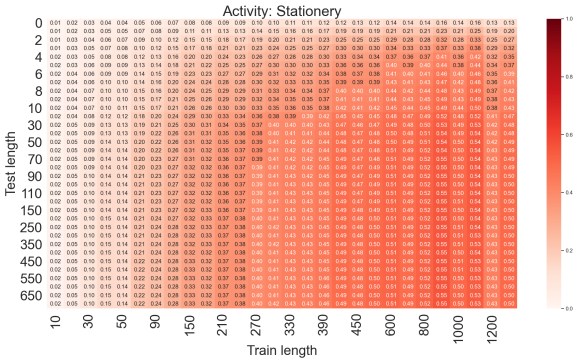
(a) For exercise, $r^{Exercise}_{L>0,l>0}$



(b) For walking, $r^{Walking}_{L>0,l>0}$

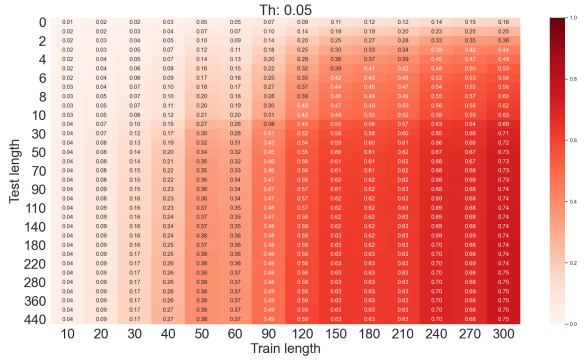


(c) For sports, $r^{Sports}_{L>0,l>0}$

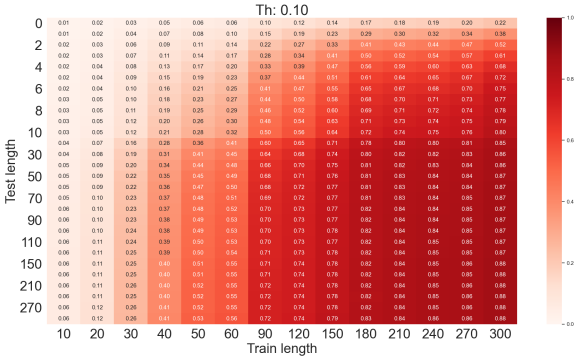


(d) For stationary, $r^{Stationary}_{L>0,l>0}$

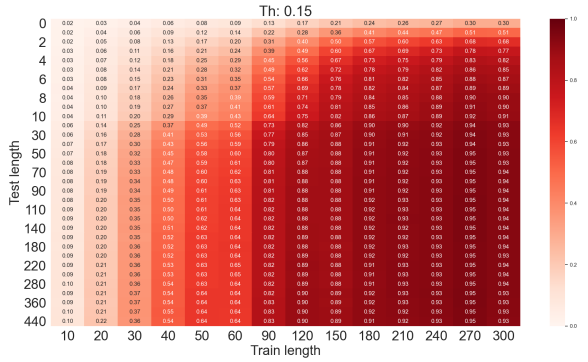
Figure 19: Effect of training and test length on re-identification risk for *Exercise, Walking, Sports*, and *Stationary*. Here, $r^{A}_{L,l}$ represents re-identification risk when train length is L, test length is l, and activity type is A. Please see Section 6.5.4 for details on re-identification risk characterization.



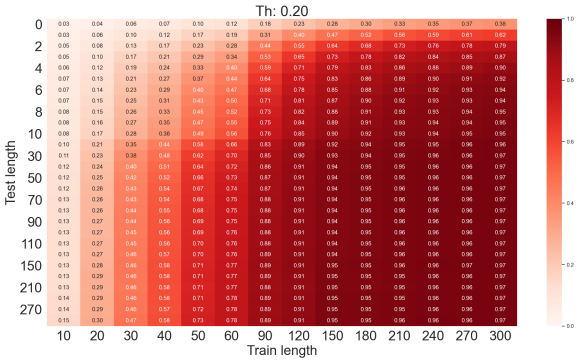
(a) For thresholding, $r^{SD>0.05}_{L>0,l>0}$



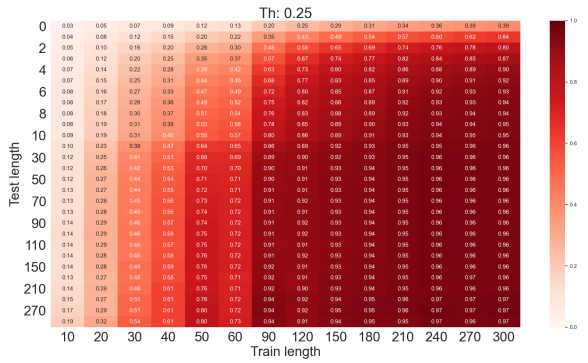
(b) For thresholding, $r^{SD>0.1}_{L>0,l>0}$



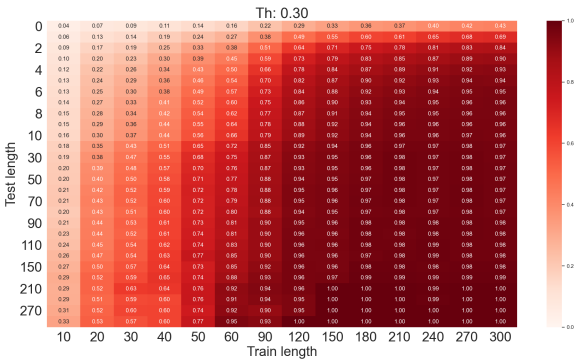
(c) For thresholding, $r^{SD>0.15}_{L>0,l>0}$



(d) For thresholding, $r^{SD>0.20}_{L>0,l>0}$



(e) For thresholding, $r^{SD>0.25}_{L>0,l>0}$



(f) For thresholding, $r^{SD>0.30}_{L>0,l>0}$

Figure 20: Effect of training and test length on re-identification risk for coarse grained activity classes based on standard deviation thresholds. We use the notation $r^{SD_{th}}_{L,l}$ to denote risk when Standard Deviation threshold (SD_{th}) is varied together with training length (L) and test length (l). See Section 6.5.4 for more details on re-identification risk.