Privacy-Preserving IoT framework for activity recognition in personal healthcare monitoring

Theo Jourdan*[1,2], Antoine Boutet†[1], Amine Bahi‡[3], and Carole Frindel§[2]

[1]Univ Lyon, INSA Lyon, Inria, CITI, F-69621 VILLEURBANNE, France
[2]Univ Lyon, INSA Lyon, CNRS, Inserm, CREATIS UMR 5220, U1206, F-69621 VILLEURBANNE, France
[3]Universit Mohammed 6 polytechnique, Ben Guerir, Maroc

Abstract

The increasing popularity of wearable consumer products can play a significant role in the healthcare sector. The recognition of human activities from IoT is an important building block in this context. While the analysis of the generated datastream can have many benefits from a health point of view, it can also lead to privacy threats by exposing highly sensitive information. In this paper, we propose a framework that relies on machine learning to efficiently recognise the user activity, useful for personal healthcare monitoring, while limiting the risk of users re-identification from biometric patterns characterizing each individual. To achieve that, we show that features in temporal domain are useful to discriminate user activity while features in frequency domain lead to distinguish the user identity. We then design a novel protection mechanism processing the raw signal on the user's smartphone to select relevant features for activity recognition and normalise features sensitive to re-identification. These unlinkable features are then transferred to the application server. We extensively evaluate our framework with reference datasets: results show an accurate activity recognition (87%) while limiting the re-identification rate (33%). This represents a slight decrease of utility (9%) against a large privacy improvement (53%) compared to state-of-the-art baselines.

1 Introduction

The increasing popularity of wearable products and the emergence of medical Internet of Things (IoT) devices have paved the way for personal healthcare monitoring at home or in hospital environments, aging and assisted living, and childcare. These devices record electronic health measurements from a variety of sensors and send these patient data to an application server to be processed and analysed. These processing and analysis include for instance advanced signal processing and machine learning algorithms to provide a variety of services such as (1) motion tracking: number

*theo.jourdan@creatis.insa-lyon.fr
†antoine.boutet@insa-lyon.fr
‡bahiaminec@gmail.com
§carole.frindel@creatis.insa-lyon.fr

of steps, burned calories, traveled distance and sleep monitoring and (2) vital signs measurement: heart rate, skin temperature, electrocardiogram (ECG) and electroencephalogram (EEG) [33]. In the particular context of monitoring patient activity, an important number of studies use data from motion sensors to quantify and define physical activity in real-word settings for different case studies. Some studies have shown that motion sensors are reliable tools to assess long-term physical activity for cancer patients during their therapy [32]. Moreover these tools could also be successfully used in the development of customized rehabilitation programs for cancer [16, 60], obesity [67] and neurological disorders [23]. These methods have the advantage of being low cost and make it hence possible to follow many patients outside the hospital.

However, due to their nature, collected data from medical IoT devices are highly sensitive. Advances in wireless communication and web technologies facilitate the remote real-time monitoring of such systems [72]. However, the complex workflow of collected medical data multiplies the security and privacy risks all along the life-cycle of the data including the data collection and transmission [6, 70], as well as the processing and the storage [58]. When such medical data can be accessed by an adversary, risks of privacy threats like leakages of sensitive information or user re-identification are very high (e.g., the re-identification of Governor William Weld's medical information [44]). In the context of activity recognition through mobile devices, the challenge is to identify data that can preserve the privacy of individuals while still being relevant enough for machine learning tasks [61]. This challenge raises two important questions: 1) Is the collected data protected enough so that no one can misuse it to infer sensitive information or to re-identify the owner? 2) How to assess whether the protected data are still accurate enough for researchers in the health domain (especially if these researchers still require enough information to test and evaluate new or evolving programs and protocols)? Achieving this balance between data utility and data privacy is an important objective to send secure and reliable data through mobile devices and to strengthen end-user confidence and adoption.

In this paper, we propose a privacy-preserving framework for activity recognition from mobile devices. This framework relies on a machine learning technique to efficiently recognise the user activity pattern, useful for personal healthcare monitoring, while limiting the risk of re-identification of users from biometric patterns that characterizes each individual. In this context, re-identification is associated with the ability of a machine learning model to determine how different one user data is from other users [35, 53]. To achieve that, firstly we extracted well-known multiple features [62, 63] from raw signal and deeply analysed their impact on both the activity recognition and the user re-identification. We show that features in temporal domain are useful to discriminate user activity while features in frequency domain lead to discriminate the user identity.

Based on this observation, we design a novel privacy-preserving framework. In this framework, data records are processed locally on the user device and only relevant features are extracted. Additionally, features in the frequency domain (i.e., features leading to discriminate users) are normalized. This normalization can be viewed as a generalization-based approach. However compared to other generalization-based approaches based on k-anonymity that are well known to drastically reduce the utility of the protected data [28], our solution keeps a high utility (i.e. activity recognition) while providing a good privacy (i.e. small user re-identification). Once normalized, this information is periodically uploaded to the application server. Each batch of features is stored independently on the server (i.e., with a different pseudonym) to avoid linking both batches to individuals and batches together. Moreover, to avoid centralizing both the data and the associated identity of their owners on the same node, the mapping between the pseudonyms and the user identities is only retained by the hospital practitioners.

We exhaustively evaluated our machine learning framework with the use of two references datasets. Results show an average accuracy of 87% in activity recognition while limiting the user re-identification rate up to an accuracy of 33%. We also compared our solutions against different baselines. Our solution provides a better privacy-utility trade-off with a slight decrease of utility (9% drop in accuracy) against a large increase of privacy (53% drop in accuracy). In addition, by processing the signals at the edge of the network on the smartphone of users, our framework drastically reduces the operational costs on the application server (a decrease of 81% for computational cost). Lastly, we assess our framework with another dataset containing signals more perturbed by noise. In this case, we show that the impact on the accuracy of our framework remains limited and mostly impacts the static activities (e.g., standing activity). However, this impact can be removed by adapting the preprocessing step with filters according to the considered signals.

Our contributions can be summarized as follow:

- We quantify both the risk assessment associated with the re-identification of users (90% in average) and the capacity to detect the user activity (97% in average) from signals from mobile devices. Knowing that the state of the art in activity recognition is almost at the same accuracy [5]
- We deeply analysed the impact of multiple features on both the activity recognition and the user re-identification. We show that features in the temporal domain tend to discriminate the user activity while features from the frequency domain tend to discriminate users.
- We propose an efficient workflow and machine learning technique to recognise user activity with high utility while limiting the risk of user re-identification. Our solution provides a better privacy-utility trade-off with a slight decrease of utility (9%) against a large increase of privacy (53%) compared to state-of-the-art baselines, while reducing the computational cost on the application server.
- We test the capacity of our approach to be generalized by showing that the privacy-utility trade-off is better regardless of the considered classifier and also by assessing our framework with another smartphone dataset containing signals more perturbed by noise. We show a limited impact on the accuracy provided by our framework and we show that this impact can be removed by adapting the preprocessing according to the considered signals.

In this paper, we present background on privacy and IoT healthcare workflow in Section 2 before we define the adversary model in Section 3. We then quantify and analyze the capacity of both recognizing the activity of users and their identity in Section 4. Section 5 details our privacy-preserving framework and Section 6 presents its evaluation. Finally, related work is reviewed in Section 7 before we conclude in Section 8.

2 Methodology

This section explains the methodology we followed for activity recognition and user re-identification using IoT mobile devices. Although this description is specific to our methodology, it is typical and provides background on IoT healthcare workflow. The whole workflow is depicted in Figure 1 and includes data acquisition (Section 2.1), signal preprocessing (Section 2.2), segmentation (Section 2.3), feature extraction (Section 2.4), and classification (Section 2.5). Figure 1 also shows that the purpose (i.e., the activity recognition) and one privacy risk (i.e., user re-identification) are made through a common pipeline. These two tasks are done on the basis of classification with joint approaches (descriptors and machine learning algorithms). Section 3 provides more details about the privacy risk assessment and the considered adversary model.

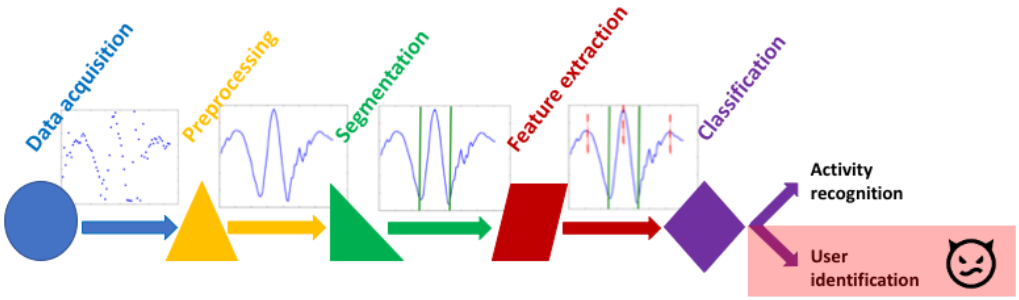


Figure 1: Traditional IoT healthcare workflow for activity recognition, an adversary can misuse the classifier to re-identify users.

2.1 Data acquisition

Data acquisition relies on sensors that are present in IoT devices, such as smartphones, smartwatches, smart wristbands, tablets and medical sensors. There exist a variety of sensors that allow the acquisition of various types of data, which can then be used for different types of tasks. For the recognition of physical activities, the authors in [56] propose to use of inertial sensors, i.e., accelerometers and gyroscopes, complemented with orientation measurement using magnetic sensors, e.g., a compass and a magnetometer, and location measurement using location sensors, e.g., a Global Positioning System (GPS).

The data acquisition process is accomplished by a specific module in the mobile device and consists of the measurement and conversion of the electrical signals received by each sensor into a readable format [59]. Several challenges are associated with the data acquisition process when recognizing physical activities, including the positioning of the mobile device, the data sampling rate and the number of sensors to be used and hence managed [10]. All these factors directly influence the correct extraction of meaningful features. As the sensors are embedded in the mobile device, they cannot be located separately in different parts of the body; rather, the mobile device needs to be situated in a usual and comfortable position. Another issue related to mobile devices is the power consumption of the data acquisition tasks. Multitasking execution patterns differ among mobile devices, because these depend on their processing ability, memory and power capabilities and on the operating system and on the number and type of mobile applications currently installed and/or running. The selection of the best data acquisition methods depends on the purpose of use, the type of data acquired and their environment [22, 55].

2.2 Signal preprocessing

Sensor signals are typically preprocessed by the application of a series of filters. First, noise was reduced with a median filter and a third order low-pass Butterworth filter with a cutoff frequency of 20 Hz. This frequency threshold was selected from the work presented in [39] which states that the energy spectrum of the human body motion is below 15 Hz. The resulting signals were further filtered to break them down into channels that make sense from a physical point of view as displayed in Figure 2. For example, linear acceleration signal was decomposed in two principal channels: gravitational and body motion components. This step was performed using another low-

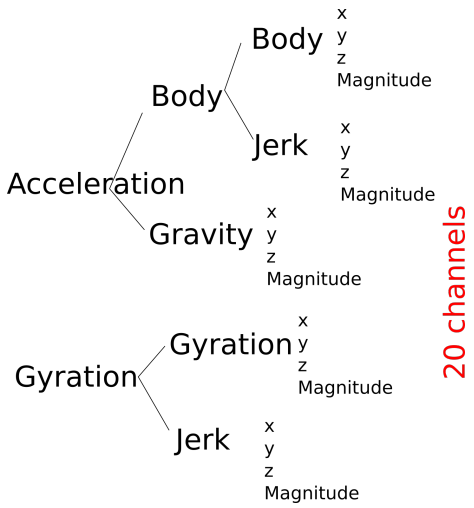


Figure 2: Channels considered for feature extraction.

pass filter and assuming that the gravitational component mainly refer to the lowest frequencies [5]. Subsequently, body motion acceleration and gyration signals were derived in time to obtain jerk that reflect the temporal variations of the signals. Finally, signals were decomposed according to their acquisition axes (x, y, z, respectively) in order to observe them in a specific direction (vertical, lateral or longitudinal) as depicted Figure 3. The magnitude of associated signals has also been calculated to produce an average signal less sensitive to how the device is fixed on the person. This filtering step allowed us to reach 20 channels in total.

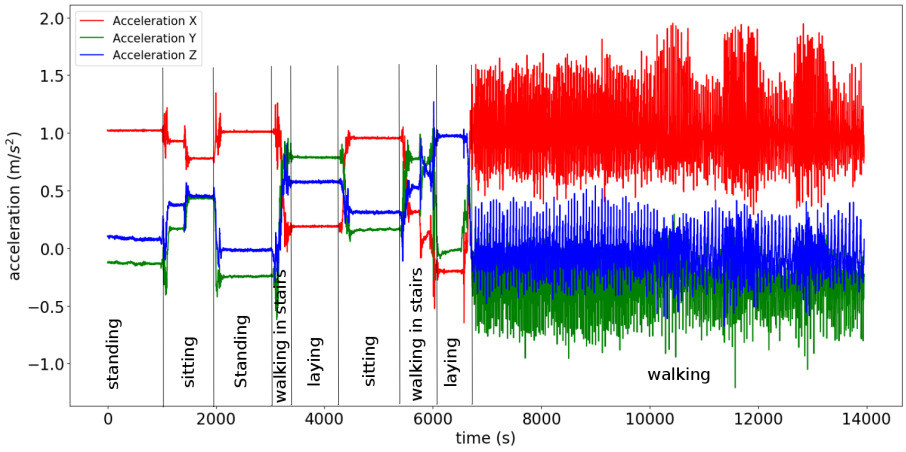


Figure 3: Visualization of accelerometer signals in x, y and z dimensions and associated activities.

2.3 Segmentation

Channel signals are typically segmented using a fixed sliding window technique. Windows with a span of 2.5 seconds and an overlap of 50% were captured. An overlap degree of 50% means that the window is shifted by half of its size, in other words 50% of the previous data are included in the next window. The choice of the window size is not trivial especially for an activity recognition algorithm. A small window size could split an activity signal while large window size could contain multiple activity signals. We decided to calibrate our window size on the most complex activity: walking. Hence, the window size has been chosen to take into account at least a full walking cycle of two steps: the cadence range of an average person walking corresponds to minimum speed of 1.5 steps by second according to [9].

	Function	Description	Formulation
Time domain	mean $(\boldsymbol{s})$	Arithmetic mean	$\bar{s} = \frac{1}{N}\sum_{i=1}^{N} s_i$
	std $(\boldsymbol{s})$	Standard deviation	$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(s_i - \bar{s})^2}$
	mad $(\boldsymbol{s})$	Median absolute deviation	$\text{median}_i\left(\lvert s_i - \text{median}_j(s_j)\rvert\right)$
	max $(\boldsymbol{s})$	Largest values in array	$\max_i(s_i)$
	min $(\boldsymbol{s})$	Smallest value in array	$\min_i(s_i)$
	sma $(\boldsymbol{s}_1, \boldsymbol{s}_2, \boldsymbol{s}_3)$	Signal magnitude area	$\frac{1}{3}\sum_{i=1}^{3}\sum_{j=1}^{N}\lvert s_{i,j}\rvert$
	iqr $(\boldsymbol{s})$	Interquartile range	$\text{Q3}(\boldsymbol{s}) - \text{Q1}(\boldsymbol{s})$
	autoregression $(\boldsymbol{s})$	4th order Burg Autoregression coefficients	$\boldsymbol{a} = \text{arburg}(\boldsymbol{s}, 4), \boldsymbol{a} \in \mathbb{R}^4$
	correlation $(\boldsymbol{s_1}, \boldsymbol{s_2})$	Pearson Correlation coefficient	$C_{1,2}/\sqrt{C_{1,1}C_{2,2}}, C = \text{cov}(\boldsymbol{s}_1, \boldsymbol{s}_2)$
	angle $(\boldsymbol{s}_1, \boldsymbol{s}_2, \boldsymbol{s}_3, v)$	Angle between triaxial signal mean and vector	$\tan^{-1}\left(\lVert[\bar{s}_1, \bar{s}_2, \bar{s}_3] \times \boldsymbol{v}\rVert, [\bar{s}_1, \bar{s}_2, \bar{s}_3] \cdot \boldsymbol{v}\right)$
Frequency domain	skewness $(\boldsymbol{s})$	Frequency signal Skewness	$\text{E}\left[\left(\frac{\boldsymbol{s}-\bar{s}}{\sigma}\right)^3\right]$
	kurtosis $(\boldsymbol{s})$	Frequency signal Kurtosis	$\frac{\text{E}\left[(\boldsymbol{s}-\bar{s})^4\right]}{\text{E}\left[(\boldsymbol{s}-\bar{s})^2\right]^2}$
	maxFreqInd $(\boldsymbol{s})$	Largest frequency component	$\arg\max_i(s_i)$
	energy $(\boldsymbol{s})$	Average sum of the squares	$\frac{1}{N}\sum_{i=1}^{N} s_i^2$
	entropy $(\boldsymbol{s})$	Signal Entropy	$\sum_{i=1}^{N}(c_i \log(c_i)),\ c_i = s_i / \sum_{j=1}^{N} s_j$
	meanFreq $(\boldsymbol{s})$	Frequency signal weighted average	$\sum_{i=1}^{N}(i s_i) / \sum_{j=1}^{N} s_j$
	energyBand $(\boldsymbol{s}, a, b)$	Spectral energy of a frequency band $[a, b]$	$\frac{1}{a-b+1}\sum_{i=a}^{b} s_i^2$

Figure 4: List of measures for computing feature vectors. N: signal vector length, Q: quartile.

Xacc_body_iqr	Xacc_body_max	Xacc_body_mean	Xacc_body_med	Xacc_body_min	Xacc_body_ropy	Xacc_body_std		pers	act
0.77666792327	1.01481659060	0.32071585656	0.34988767835	-0.49054102707	4.7489565343	0.4194744014		10	2
0.66693512370	1.43263647481	0.26841672908	0.43411692212	-1.41238613704	4.7722314297	0.6610481443		10	3
1.02907915173	1.43263647481	-0.10075092775	0.08232560553	-1.42686548654	4.7899910551	0.6334978541	...	10	3
0.23557396729	0.74911155782	0.33652443467	0.26582976888	0.10360618631	4.8056637254	0.1568877474		10	4
0.35584093169	0.78654658654	0.21654485464	0.27026656791	-0.72443435405	4.7194139887	0.3592030590		10	4

Figure 5: A sample dataset with features and labels, input of the classification step.

2.4 Feature Extraction

From each window of each channel signal, a feature vector was extracted which contained 17 measures estimated in the time and frequency domains respectively. The Discrete Fourier Transform (DFT) was used to extract the descriptors of each window in the frequency domain. The choice of these descriptors was made on the basis of an earlier review on effective descriptors for gait recognition [63] : e.g. for time domain mean, standard deviation (STD), signal magnitude area (SMA) and signal-pair correlation (Corr); and for frequency domain energy and entropy. The selected measures to obtain the feature vector are depicted in Figure 4. A feature vector was calculated from each experiment window sample and labeled according to the user and activity it belongs. Figure 5 shows an example of the dataset format, where lines correspond to window samples and columns to features (except the two last ones which correspond to the labels). Such dataset is used as an input for the classification task. A total of 340 features (20 channels x 17 measures) are extracted. The notation for naming a descriptor in the rest of this article is the following $\{orientation\}\_\{channel\}\_\{descriptor\}$.

2.5 Classification

Machine learning algorithm There are multiple machine learning algorithms that can effectively handle these features (e.g., Decision Tree, Support Vector Machine, Random Forest). We evaluated a number of them (one representative for each machine learning family) as illustrated in Table 1. In order to make a fair comparison, the different algorithms were optimized independently. From this analysis, it follows that it is Random Forest (RF) that provided the best results for our use case. Consequently, RF was chosen for the multi-class classification tasks in the remainder of this study. In general, the RF algorithm is a supervised classifier having fast training time and very high performance without fine-tuning [50]. The function "RandomForestClassifier" of the Python Scikit Learn package [51] was used to build the RF classifier and related to its optimization, 700 was chosen as the number of trees in the forest, $\sqrt{n}$ random features were considered in building each tree and 10 was set as the maximum depth of each tree.

Algorithm	**Accuracy (activity)**	**Accuracy (identity)**
Decision Tree	0.94	0.73
K-nearest Neighbors	0.78	0.36
Support Vector Machine	0.58	0.23
Gaussian Naive Bayes	0.80	0.14
Random Forest	0.96	0.82
Quadratic Discriminant Analysis	0.88	0.63

Table 1: Comparison of different well-known algorithms in terms of activity and identity performance.

Utility and privacy measures To measure the classification quality based on the proposed features with RF, we computed the accuracy from the confusion matrix [34]:

$$Accuracy = \frac{|TP| + |TN|}{|TP| + |TN| + |FP| + |FN|},$$

where $|TP|$ (True Positive): is the number of correct predictions for a specific event value, $|TN|$ (True Negative): is the number of correct predictions for non-event values, $|FP|$ (False Positive): is the number of incorrect of predictions for a specific event value, and $|FN|$ (False Negative): is the number of incorrect predictions for non-event values.

Accuracy reflects the number of correct predictions made by the model over all kinds of predictions made. Accuracy is comprised in $[1:0]$ where a value of 1 corresponds to a perfect prediction. We prefer the accuracy rather than the f-score because the variable classes in the data are nearly balanced. We use this metric to compute the quality of our classification to predict both the activity of the user and the user identity. We leverage this metric to define a utility and a privacy measurement. Specifically, we called *Accuracy(activity)* the result of the accuracy when it is applied to the activity recognition (utility metric), and we call *Accuracy(re-identification)* the result when it is applied to the user identity (privacy metric).

Algorithm 1: Feature selection

Input : List of features sorted by importance f and associated initial accuracy a; $threshC = 0.7$; $threshA = 0.03$
Output: List of selected features

```
1  for each feature f_i ∈ f do
2      Compute the Pearson correlation values C for each feature in {f − f_i} : fcorre
3      for each feature f_j ∈ fcorre do
4          if | C(f_j) |> threshC then
5              Compute accuracy newA of classification for {f − f_j} : newa
6              if a − newA < threshA then
7                  Erase feature f_j from f
8              end
9          end
10     end
11 end
```

Feature ranking and selection The RF algorithm can also be used to rank features according to their importance in the classification. When training a tree, it can be computed how much each feature decreases the Gini impurity index [38] in a tree. For a forest, the impurity decrease from each feature can be averaged and the features are ranked according to this measure.

The RF algorithm can also be used for feature selection [14]. This is done via measuring the mean decrease of accuracy when a particular feature is removed from the set of features in the trees. If the accuracy deterioration after feature exclusion is negligible, the feature is less important and vice versa. The importance scores of the features in the RF classifier [14, 29] can therefore be evaluated and used as a feature selection criteria. For more details, see the Algorithm 1: It consists of two nested loops, one corresponding to features ranked by importance (line 1) and one corresponding to features correlated to each of the features of the first loop (line 3). The correlation is calculated using the Pearson coefficient (line 2). If the correlation between two features is greater than a certain threshold (line 4), then the accuracy of the random forest algorithm is recalculated after removal of the correlated feature (line 5) and if the corresponding decrease in accuracy is below a certain threshold (line 6) this feature is eliminated for good (line 7).

3 Adversary model

This section presents the architecture of a traditional centralized system without any protection and the potential attack we want to protect the system from (Section 3.1). Then we present the assumptions made to design our solution (Section 3.2).

3.1 Traditional architecture

In this traditional architecture depicted Figure 6, (❶) IoT devices or directly the smartphones perform the data collection from sensors and (❷) send the raw data to the application server which stores them. (❸) The server then performs all the remaining tasks including the preprocessing, the segmentation, feature extraction, and the classification of the activity. Finally, (❹) the hospital practitioner requests the application server to have an analysis of the activity of patients.

This centralization of the raw data exposes users to many privacy risks in case of data leak. Indeed, if the server is compromised or if some data are stolen, raw data are revealed leading to the possibility to do many sensitive inferences including re-identification. In this work, we focus our privacy assessment on this user re-identification risk.

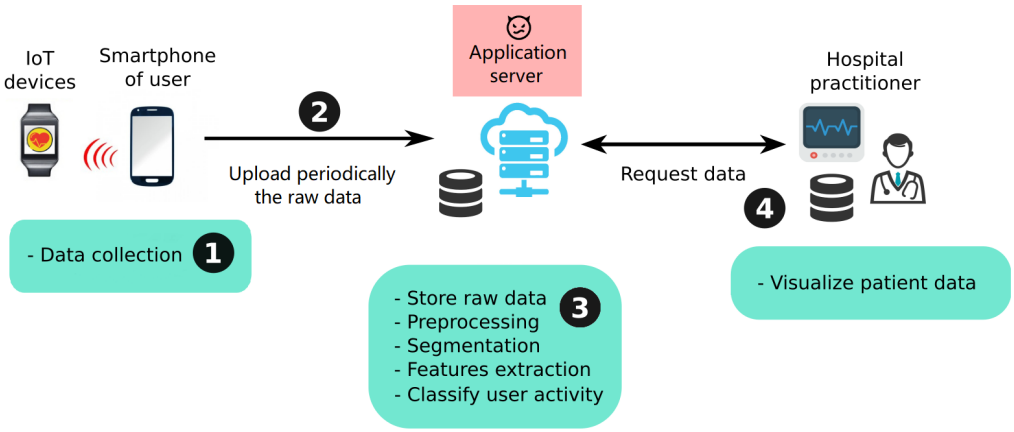


Figure 6: A traditional architecture: the user smartphone send directly the raw data to the application server that upload it periodically.

3.2 Technical assumptions

Before presenting our privacy-preserving framework in Section 5, we describe our assumptions and the adversary model against which our solution is designed. The framework presented in this paper involves three premises: the client running on the smartphone of users, the application server storing the features and performing the classification, and the hospital practitioner monitoring the patient activity. First, we assume that the client application and the smartphone on which it is run are trusted. This means that the data acquisition, the preprocessing, the segmentation, the feature extraction, and the normalization cannot deviate from a correct behaviour. Moreover, we do not consider limitations on the sampling rate of the data acquisition as in [65].

Second, we assume that the application server runs on public cloud platforms. We consider that this cloud platform is honest but curious [27]. This means that the application server behaves correctly when it comes to processing data received from clients. More precisely, this means that the data is stored correctly in the database, that no forged information can be injected in the database, and that the classifier model cannot be maliciously tampered. However, we assume that the adversary is able to collect part or the entire information stored in the database. Each information corresponds to independent batches of data unlinked to users (i.e., with a different random pseudonym for each batch). Additionally, we assume that the adversary is able to collect data relative to the gestures of each user from a malicious IoT device for instance. This prior knowledge on each user is used by the adversary to build a classifier model. This classifier exploits the same preprocessing, segmentation, and features than our classifier but with the objective to predict the identity of the user for each batch of data stored in the database.

Third, we assume that the server used by the hospital practitioner is trusted. This server is used to store the mapping between the batches of data sent to the application server and the identity of the users.

Lastly, all communications between nodes (i.e., clients, the application server, and server of the hospital practitioner) are secured. We assume that no information can be inferred from these secured communications.

4 Quantifying activity recognition and user re-identification

We carried out an extensive evaluation of the capacity to recognise the activity of users and to re-identify them. We show that following the methodology described in Section 2, we are able to predict the activity of the user with a very high rate of success. In addition, we show that without any protection scheme, data from mobile devices act as a personal fingerprint and lead to re-identify users. We first describe the dataset used in this evaluation (Section 4.1) before to quantify the activity recognition and the user re-identification (Section 4.2). Finally, we analyse the impact of extracted features (Section 4.3).

4.1 Dataset

The dataset used in this work is available online for public use as the "Human Activity Recognition using Smartphones" dataset in the UCI Machine Learning Repository [5]. This dataset represents a reference for evaluating activity recognition learning models. It is composed of the 3-axial raw data from accelerometer and gyroscope sensors read at a constant frequency of 50 Hz. A group of 30 volunteers were selected to follow a protocol of activities while wearing a smartphone on their waist. The experiment was planned in order to contain six basic activities: three static postures (standing, sitting, lying-down) and three ambulation activities (walking, walking-downstairs and walking-upstairs). Figure 3 displays accelerometer signal of one of the experiments and the associated activities. The protocol of activities is detailed in [57]. The duration of an entire experiment was around 15 minutes and was repeated ten times. All the experiments were recorded on video to have a ground truth to annotate the performed activities on acceleration and gyration signals.

Activity	Accuracy(activity)
Walking	0.97
Walking upstairs	0.95
Walking downstairs	0.94
Sitting	0.97
Standing	0.98
Laying	0.99

Table 2: User activities can be recognised with a high success rate (recognition using the methodology presented Section 2).

4.2 Activity Recognition and User Re-Identification

We firstly evaluated the accuracy of different well-known classification schemes for the activity recognition and the user re-identification in order to select the best one for our use case (Table 1). Without optimizing parameters (i.e., using standard values), RF outperforms other schemes for both classification tasks with 0.96 and 0.82 of accuracy for activity recognition and user re-identification, respectively. Once the most adapted classification scheme identified, we then optimized parameters to further increase the accuracy.

Table 2 summarizes the accuracy for the recognition of the different activities. Results show that our machine learning framework is able to highly recognise activities with an average accuracy of 0.97 which is comparable to state-of-the-art performance [40]. As the table indicates, the accuracy is lower for ambulatory activities in stairs. A possible explanation for this is that these activities correspond to the smallest acquisition times (Figure 3).

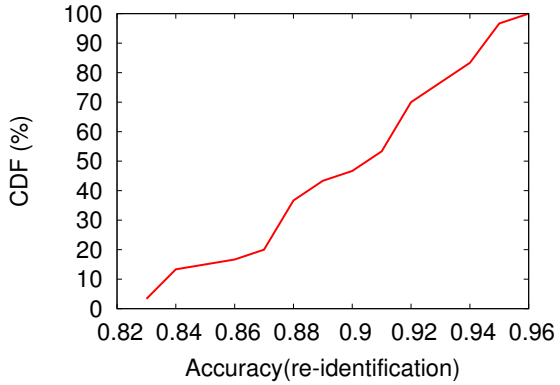
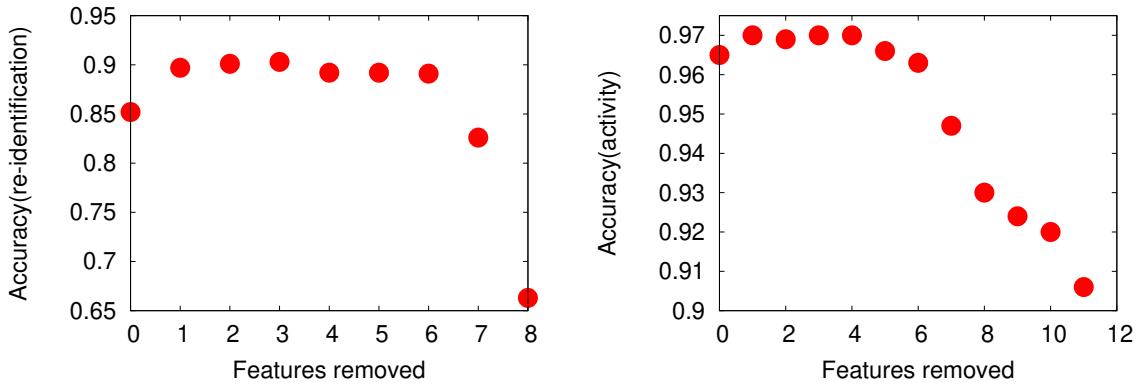


Figure 7: Cumulative distribution of the accuracy for the user re-identification task: users can be easily re-identified from their data.

Figure 7 depicts the cumulative distribution of the accuracy for the user re-identification task. Accuracy ranges from 0.82 to 0.96 among the 30 users with an average of 0.90. These results indicate that the data collected from the gesture of users characterizes each individual and can lead to re-identify them with a high success rate. However, the task of re-identification is slightly more difficult than that of recognizing activities with lower accuracy.



(a) Privacy (b) Utility

Figure 8: Impact of the number of features (depicted in Table 3 and Table 4) retained in the RF learning process on user's privacy and utility metric (features were sorted by increasing order of importance).

4.3 Impact of Features

The previous experiments are also used to rank features (from the 340) according to their importance. Eight and eleven features were respectively selected for the activity recognition and user re-identification tasks given the correlation and accuracy analysis (see Algorithm 1 for methodology and Tables 3 and 4 for results) with one temporal feature in common (Magn_grav_max). Indeed, many features are alike and contain similar information on the original sensor data. Compared to using all 340 features, using only these 19 relevant features lowers only slightly ($< 4\%$) the two classification tasks performance (97% vs 96% for activity classification and 90% vs 86% for user re-identification). This can be observed more precisely in the Figures 8a and 8b, where the importance of each selected feature is independently tested for the task of interest: there is a strong correlation between the importance of a specific feature and the performance of the RF algorithm after removing it.

Based on these ranking results, it is interesting to note that the task of activities recognition (i.e., utility) is almost exclusively (9 of the 11 selected features) operated in the time domain whereas the task of user identification (i.e., privacy) is based (5 of the 8 selected features) on features in the frequency domain. These results can be explained by the fact that the activities are mainly distinguished from each other by their level of amplitude in acceleration and gyration (Figure 3) and therefore their associated statistics. Conversely, the user identification is more related to the pace or cadence at which this person performs the activity and is strongly related to biomechanics (e.g., age, size, weight).

5 Privacy-Preserving Activity Recognition Framework

To ensure privacy, our framework relies on both an architecture limiting the exposure of sensitive information and a data normalisation applied on features leading to re-identify users (Section 4.3).

Features	Importance
Y_grav_std	0.175
Z_grav_med	0.163
Z_grav_energy	0.137
X_grav_max	0.128
Magn_grav_max	0.123
Y_gyro_mean	0.107
Y_gyro_irq	0.088
Y_body_zcross	0.079

Table 3: Most important features for user re-identification (frequency-based features are in grey).

Features	Importance
X_grav_max	0.144
X_grav_min	0.127
Magn_grav_max	0.109
X_gyro_min	0.104
X_body_var	0.098
Magn_body_var	0.085
X_gyro_max	0.082
Y_gyro_irq	0.078
X_gyro_mean	0.077
Magn_gyro_mean	0.074
Y_body_entropy	0.020

Table 4: Most important features for activity classification (frequency-based features are in grey).

These normalisations act as a form of generalisation-based obfuscation. In this section, we first present the architecture of our framework (Section 5.1) before to describe the normalisation of each sensitive feature (Section 5.2).

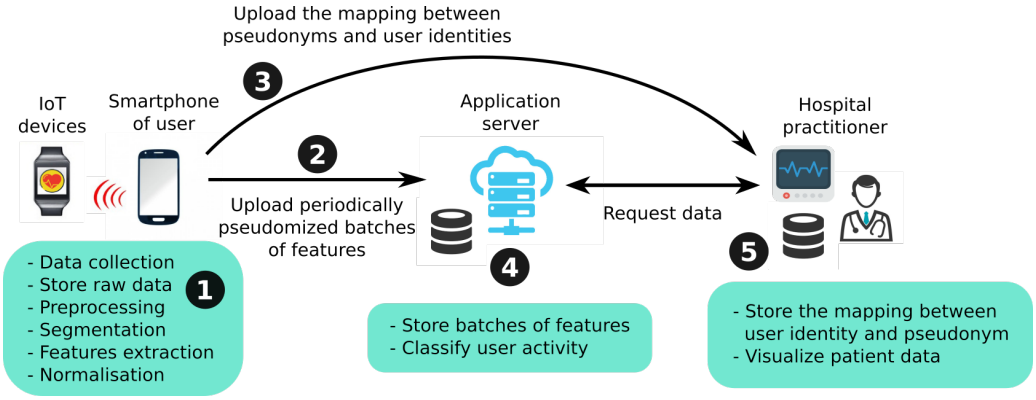


Figure 9: Architecture of our framework: the user smartphone is leveraged to extact relevant features and only these features are uploaded periodically to the application server.

5.1 Architecture

The design of our privacy-preserving framework comprises three main elements: a client application running on the user smartphone communicating with its IoT environment, the application server, and the hospital practitioner. To limit the exposition of sensitive information, the application server does not store identified data but only batches of features where each batch is randomly pseudo-anonymized. Only the hospital practitioner retains the mapping between the user identities and pseudonyms, and requests the application server to monitor the activity of users.

The architecture of our privacy-preserving activity recognition framework is depicted Figure 9. Firstly, (❶) IoT devices (e.g. smartwatch) or directly the smartphones perform the data acquisition. In both cases, these raw data are stored locally on the smartphone. The client application then performs the preprocessing, the segmentation and the features extraction following the methodology described in Section 2. On the basis of our analysis on the importance of features, this feature extraction only concerns the 19 features identified as important (Section 4.3). Moreover, the client conducts the normalisation of the features identified as leading to the re-identification of users. All these normalisations are described in the following sub-section. As all the aforesaid actions performed on the smartphone only concern the associated user on one batch of data (i.e., one day for instance), the resulting computational cost is cheap (Section 6.3). Secondly, (❷) the client application associates a random pseudonym to each timestamped batch of features before to periodically upload them to the application server. (❸) The client application then sends to the hospital practitioner the list of pseudonyms associated to its identity.

(❹) When a batch of features is received by the application server, it stores this information in a database. Consequently, each batch in this database does not contain the identity of the user but a random pseudonym. The application server then periodically performs the classification to detect the activity associated with each batch of features.

Finally, when the hospital practitioner wants to monitor the activity of a specific user, firstly it retrieves locally all the pseudonyms associated with the specified user and then requests the application server to have the activity history of the specified pseudonyms (❺). In this approach, the requests from hospital practitioners could lead to link different pseudonyms to the same patient. To overcome this problem, fake requests could be sent to hide the ones targeting the expected patient.

5.2 Normalisation

In order to limit the re-identification of users, we propose a normalisation approach which generalises the effect of the different descriptors identified as important for the task of user re-identification. Similar to k-anonymity presented Section 7.1 which ensures anonymity group gathering k different users with the same quasi-identifier, our normalisation approach aims at erasing the characteristics of a single specific user (i.e., leading to the re-identification) and transforming the data so that, after normalisation, the data of all users share the same statistical characteristics. Given the data from the sensors noted S and of size n, applying the normalisation approach on S will output the so-called "normalised data" noted S^*. In this work, we distinguished five normalizations, each of them referring to the features in the frequency domain listed in Table 3 (the feature corresponding to the normalisation is given in parentheses). For the three temporal features that remain in Table 3, we proposed to delete the two of them that were not used for activity recognition. We kept the last one because it was also selected for activity recognition in Table 4.

5.2.1 Normalisation by mean

(Y_gyro_mean)

$$S_i^* = S_i - \mu + \mu^*, \quad i \in [0, n], \quad \mu^* = 0, \tag{1}$$

with μ and μ^* being respectively the data means before and after normalization.

5.2.2 Normalisation by interquantile range

(Y_gyro_irq)
The interquantile range (IQR) is a measure of statistical dispersion, being equal to the difference between 75th and 25th percentiles.

$$S_i^* = \frac{S_i}{IQR} IQR^*, \quad i \in [0, n], \quad IQR^* = 1, \tag{2}$$

with IQR and IQR^* being respectively the data interquartile ranges before and after normalisation.

5.2.3 Normalisation by standard deviation

(Y_grav_std)

$$S_i^* = \frac{S_i}{\sigma} \sigma^*, \quad i \in [0, n], \quad \sigma^* = 1, \tag{3}$$

with σ and σ^* being the data standard deviations before and after normalisation.

5.2.4 Normalisation by root mean square

(Z_grav_energy)

$$S_i^* = \frac{S_i}{\sqrt{\frac{1}{n} \sum_{j=1}^{n} S_j^2}}, \quad i \in [0, n]. \tag{4}$$

5.2.5 Normalisation by maximum and minimum

(X_grav_max)

$$S_i^* = (S_i - Min) \frac{newMax - newMin}{Max - Min} + newMin, \quad i \in [0, n], \quad newMax = 20, \quad newMin = 0, \tag{5}$$

with Max and Min being respectively the maximum and minimum of the original data and $newMax$ and $newMin$ the maximum and minimum of the normalised data.

6 Evaluation of our framework

We carried out an extensive evaluation of our framework. In this section, we start with a description of the comparison baselines (Section 6.1) before evaluating the performance of our approach in terms of utility-privacy trade-off (Section 6.2).

6.1 Comparison Baselines

To highlight the benefits of our approach, we compare the performance of our framework with that of three alternatives. The first alternative follows a perturbation scheme. Similarly to the differentially private approach described in [4] that applies a perturbation scheme in the frequency domain of aggregated time series in the context of location privacy, this alternative (called *perturbation*) adds a Gaussian noise in the signal in frequency domain before the extraction of features. The second alternative is based on simply the removal of features identified as leading to the user re-identification (Section 4.3). The incentive behind this alternative (called *suppression*) is that without these features, the re-identification is harder. The last alternative is a privacy-preserving classification based on homomorphic encryption. This alternative implements a random forest classifier working over encrypted data similar to [11]. In this solution (called *homomorphic*) the input data (i.e., the features used by the random forest model used in Section 4.2) are encrypted by the smartphone before to be sent to the server which is able to do the classification of the activity directly from these encrypted data. To achieve that, the multiple decision trees of the random forest classifier are expressed as a polynomial P whose output is the result of the classification. More precisely, each node in the trees is a boolean variable defined at 1 if, on input x, one should follow the right branch, and 0 otherwise (i.e.g, the value of a variable b_1 is 1 if the input x_1 is smaller than the threshold w_1, and 0 otherwise). Consequently, P is a sum of terms, where each term t corresponds to a path in a tree from root to a leaf node c. A term t evaluates to c only if an input x is classified along that path in the tree, else it evaluates to zero. Hence, the term corresponding to a path in the tree is naturally the multiplication of the boolean variables on that path and the class at the leaf node. We use TFHE [3] to implement this solution in C++ and the value of the inputs as well as the threshold are coded in 16 bits.

6.2 Privacy Improvement

Figure 10 reports for our solution and the baseline approaches the trade-off between the utility captured by the accuracy to recognise the activity and the privacy captured by the accuracy to re-identify users. For the baseline based on the suppression of features, each point of the curves corresponds to the deletion of a feature (from the 8 selected ones for the re-identification task). For the baseline based on perturbation, in turn, each point refers to the addition of an increasing fixed amount of noise (noise is centered on zero and its standard deviation is, for each point, increased by 2). Finally, in our framework, each point corresponds to the normalization of a growing number of features (in order of increasing importance).

Results show that the suppression approach (slope: 0.12) seems the most advantageous in terms of compromise between utility and privacy. However it is very quickly limited by the number of selected features and therefore in privacy and utility metrics; for instance the best obtained performance are respectively 0.66 and 0.93. The perturbation approach (slope: 0.34) is very effective in loss of identification however at the cost of a very important loss of utility too, with for best performance in privacy and utility metrics respectively 0.51 and 0.84. Our approach is between the two (slope: 0.21) and provides the best utility and privacy trade-off (respectively 0.87 and 0.33). Our approach based on normalization gives a better control on the weight of each feature in the protection, unlike the suppression approach which limits their impact to consideration or not.

Lastly, we also considered an adversary that trains a classifier only with features leading to the re-identification (Table 3), in this case the accuracy in terms of re-identification is less efficient than with our framework (0.17).

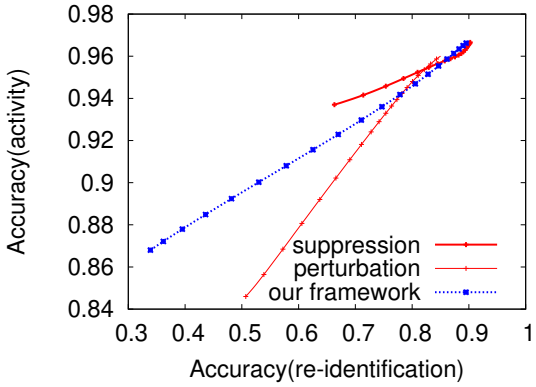


Figure 10: Our framework provides a better utility and privacy trade-off than baseline approaches.

6.3 Cost Improvement

We now compare the cost of running our framework with a traditional centralized solution based on an application server processing all the data. As described Section 3.1, in a such solution, all the data collected by the IoT devices are sent to the application server which performs all the operations including signal preprocessing, segmentation, feature extraction, and classification as described Figure 6. In comparison, our framework leverages the users smartphone to perform the signal preprocessing, the segmentation and the feature extraction, leaving to the application server only the classification task, it significantly reduces the computational cost of this server.

We measured the time spent by the application server for both the traditional approach and our approach. For our dataset (i.e., 30 users and 15 minutes of data per user), the application server spent almost 52 seconds to perform all processing against almost 10 seconds with our framework (i.e., classification only). That represents a time reduction of 81%. With a large number of users, this time reduction drastically saves the resources needed for operating an application server. Considering an application server running on a cloud infrastructure such as Amazon EC2 services [1], this reduction in terms of computational cost highlights the economic advantage of our framework.

We also measured the time spent by the application server when a homomorphic encryption scheme is used. Results show that using such a scheme to protect data generates important computational overhead. The added time introduced by the computation of the preprocessing (versus classification only) is negligible compared with this overhead.

Lastly, we evaluated the cost of running our framework on the user machine. A user only affords the cost of its activity by taking into account the preprocessing, the segmentation and the feature extraction on its smartphone. On a commodity computer, these operations applied on all the data of one user spent in average 2.5 seconds. This cost (paid every 15 minutes) remains low. In addition, these processing can be scheduled during the night when the user is inactive.

6.4 Impact of ML Scheme

To assess the generalization of results obtained by our approach with Random Forest to other classifiers, we evaluate the utility-privacy trade-off with the classifiers considered in Section 2.5 (see Table 1). Table 5 reports the privacy improvement and the utility loss of our approach according

to the considered classifier. Results depicted a similar behavior with an important decrease of the re-identification while maintaining the high level of activity recognition. There is an exception for Gaussian Naive Bayes classifier where the re-identification accuracy was already low before normalization. We also observe an increase of activity recognition after normalization for Support Vector Machine classifier probably due to the selection of features that significantly reduced the complexity of the data dimensionality that was too high on raw data for this basic classifier (i.e., we go from 340 features to 19 features after application of our approach).

Algorithm	**Utility loss**	**Privacy improvement**
Decision Tree	-0.13 (0.81)	+0.54 (0.19)
K-nearest Neighbors	-0.07 (0.71)	+0.274 (0.09)
Support Vector Machine	+0.10 (0.68)	+0.16 (0.07)
Gaussian Naive Bayes	-0.16 (0.64)	+0.077 (0.06)
Random Forest	-0.09 (0.87)	+0.49 (0.33)
Quadratic Discriminant Analysis	-0.32 (0.56)	+0.555 (0.08)

Table 5: The tendency of our approach to drastically improve the privacy while maintaining the utility is generalized to other classifiers.

6.5 Impact of the Dataset

We now evaluate the capacity of our approach to be used on another dataset. In the previous experiments, the data acquisition protocol was done with volunteers wearing a smartphone on their waist. In this section, we use another dataset (i.e., the "MotionSense Dataset" [47]) following a data acquisition done with a smartphone in the pocket of the volunteers. Consequently, the collected signals include more noise than in the previous dataset. Otherwise, this dataset is similar to the first one. It contains time-series data generated by 3-axial raw data from accelerometer and gyroscope sensors read at a constant frequency of 50 Hz. A group of 24 participants performed 6 different activities: dowstairs, upstairs, walking, jogging, sitting and standing. All these activities were made in 15 trials per user, with 9 long trials (i.e., around 2-3 minutes duration) and 6 short trials (i.e., around 30 seconds-1 minute duration).

6.5.1 Noise Sensibility

To quantify the noise present in a collected signal, we measured the Signal-to-Noise Ratio (SNR for short). SNR is expressed in decibels as the ratio of the mean signal (μ) to the standard deviation (σ) of the signal over a given neighborhood using the logarithmic scale [15] :

$$SNR = 10\ log(\frac{\mu}{\sigma}).$$

This ratio compares the level of a desired signal to the level of background noise considering that the noise in the signal is stationary in time. A SNR higher than 0 indicates more signal than noise. Table 6 reports the SNR for each activity for both datasets. Results show an important difference between the SNR of both datasets meaning a stronger presence of noise for all activities in the second dataset. This difference is significantly more important for statics activities (e.g., SNR at 25.4 versus 0.4 for standing activity). Indeed, the power of the desired signal for static activities is smaller, leaving more impact on noise.

Activity	Dataset 1 (smartphone on waist)	Dataset 2 (smartphone in the pocket)
Walking downstairs	5.05	3.8
Walking upstairs	6.6	0.1
Sitting	23.4	8.3
Standing	25.4	0.4
Walking	6.4	-3.0

Table 6: Signal-to-Noise Ratio (SNR) of collected signals of all activities for both datasets in decibels (dB) : the second dataset contains stronger noise on all activities, especially for static ones.

We now evaluate the impact of the presence of noise in the collected signal on the accuracy of our framework. Table 7 reports the accuracy of the activity recognition. Results show that our framework is still able to highly recognise dynamic activities (i.e., ranges from 0.79 to 0.89 of accuracy for walking and jogging activities) even if the collected data contains important levels of noise. However, the impact of noise on the signal of static activities drastically reduces the accuracy (e.g., 0.30 of accuracy for standing activity).

Activity	Accuracy (activity)
Walking downstairs	0.84
Walking upstairs	0.89
Sitting	0.16
Standing	0.30
Walking	0.80
Jogging	0.79

Table 7: Even if the collected data contains important level of noise, our framework is still able to highly recognise dynamic activities, while the impact of noise drastically reduces the accuracy for the recognition of static activities.

Figure 11, in turn, depicts the cumulative distribution of the accuracy for the user re-identification task on this second dataset. This distribution shows that users can be still re-identified from their data even if the signals are perturbed by noise but with a smaller success rate than with the previous dataset (0.90 versus 0.48 of accuracy on average for the previous and the new dataset, respectively).

The impact of this noise can be mitigated by refining the preprocessing with an additional filter. For instance, we experimented adding a SavitzkyGolay filter that smooths the collected signal and consequently increases the SNR of each activity.

6.5.2 Feature selection

To demonstrate the validity of our approach, it is important to ensure that the most important features used in the two classification tasks are mostly independent. Tables 8 and 9 lists for this new dataset the features selected for respectively the re-identification and the activity classification, ranked following their importance in the classification task. Results show a majority of features in the frequency domain lead to the re-identification and a majority of temporal features lead to the activity recognition.

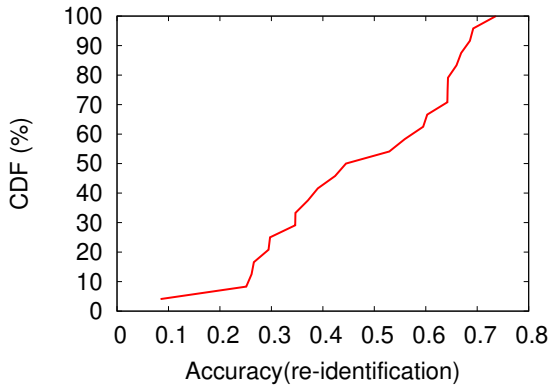


Figure 11: Cumulative distribution of the accuracy for the user re-identification task: users can be still re-identified from their data even if the signals are perturbed by noise but with a smaller success rate than with a dataset containing less noise.

Features	Importance
X_grav_mean	0.179
Y_grav_mean	0.153
Z_gyro_max	0.152
Z_grav_max	0.151
X_grav_max	0.138
Z_grav_max	0.113
Z_grav_var	0.112

Table 8: Features in the frequency domain also lead to the re-identification with the MotionSense dataset (frequency-based features are in grey)

Features	Importance
Y_gyro_mean	0.193
Y_body_std	0.184
Z_grav_std	0.167
Z_grav_mean	0.157
X_gyro_zcross	0.151
Magn_grav_min	0.148

Table 9: Temporal features also lead to the activity recognition with the MotionSense dataset (frequency-based features are in grey).

6.5.3 Framework analysis

Our approach applied on MotionSense dataset provides on average an accuracy for activity recognition of 0.97 and an accuracy of 0.37 for re-identification. These results are similar to the ones obtained with the previous dataset (0.87 of activity recognition and 0.33 of re-identification). In summary, to get the best performances from our framework, the denoising has to be designed according to the SNR of raw data before applying the feature extraction. However, our framework still provides good activity recognition while reducing the users re-identification.

7 Related work

Information privacy in the healthcare sector is an issue of growing importance. This section introduces privacy models (Section 7.1) before to present applied methods in a healthcare context (Section 7.2).

7.1 Privacy preservation models

This section introduces different common privacy preservation models. Two main privacy models have been widely adopted by the community and are still the foundation for most of subsequent works [18]. Those models propose generic privacy guarantees that are not specific to healthcare. Sensitive information can be also protected through cryptographic schemes. As these schemes are not specific to privacy, we do not present them here. However some privacy-preserving solutions relying on cryptographic-based approaches are reviewed in 7.2.

k-anonymity The idea of the model of k-anonymity [64] is to prevent one to uniquely identify individuals from a small subset of their attributes, called a *quasi-identifier*. The subset of attributes to protect, which is not part of the quasi-identifier, represent the *sensitive attributes*. For example, within medical records, the birth date, sex and zip code triplet form a quasi-identifier that is enough to uniquely identify some individuals, while the disease is a sensitive attribute. k-anonymity states that to be protected, a user must not be distinguishable among at least $k-1$ other users. To do that, all k indistinguishable users must have the same values for all attributes forming their quasi-identifier. This makes them look similar and forms an *anonymity group*. Therefore, the probability of an adversary without external knowledge to re-identify someone among k similar users is at most $1/k$.

Some weaknesses of k-anonymity have been addressed by the introduction of ℓ-diversity [46]. It extends k-anonymity by ensuring a particular distribution of values for sensitive attributes across each anonymity group. The simplest way to do so is called distinct ℓ-diversity and states that there must be at least ℓ distinct values for each sensitive field for each anonymity group. t-closeness [45] is a further extension of ℓ-diversity Instead of just guaranteeing a good representation of sensitive values, this approach enforces that the distribution of every sensitive attribute inside anonymity groups must be the same as the distribution of this attribute in the whole dataset, modulo a threshold t.

Differential privacy Differential privacy is a more recent model [20]. The idea is that an observer seeing the output of a differentially private algorithm is not able to tell if a particular individual's information was used in the computation. In other words, the addition or removal of one single element shall not change significantly the probability of any outcome of the aggregate function. Unlike k-anonymity, the differential privacy definition is not affected by the external knowledge an attacker may have.

One method to practically achieve differential privacy using numerical values relies on adding random noise following a Laplace distribution, whose magnitude depends on the *sensitivity* of the query function issued on the dataset. Intuitively, the sensitivity of a query function quantifies the impact that the addition or removal of a single element of a dataset could have on the output of this function.

Differential privacy has generated an important literature these last few years with new models and inter-model connections [18], as well as new techniques such as randomised response [68] and its combination with sampling [43] which achieves zero-knowledge privacy [24] (a privacy bound tighter than differential privacy).

Federated machine learning Federated learning became a hot research topic in recent years. In this approach, to avoid the exposition of personal data, these data remain local to the smartphone.

Instead, a learning model sent by a server is refined locally on the user device using only its local data. Information about the updated model (i.e., the weights of a deep neural network) are then sent back to the server to be aggregated. By performing these operations iteratively, the model maintained by the server converges [41, 42]. Although personal data remain local, the security guarantees are not complete due to some indirect information that may leak from the models update [54]. To overcome this problem, some researchers have proposed to leverage differential privacy in the learning process on the device or during the models aggregation [26]. Although appealing, federated learning faces several limitations. First of all, technologies are not yet mature and its deployment is not possible to all smartphones. Secondly in the context of patient monitoring, practitioners still require access to some data to test and evaluate new or evolving programs or protocols.

7.2 Applied privacy methods used in healthcare

With the technological advances of recent years, the medical domain is changing fast raising important privacy issues. For instance, new high throughput DNA sequencing technologies have drastically reduced the price and democratized DNA analysis. Due to the highly sensitive nature of this data, an important research area has emerged to address the quantification of the risk associated with this information and to protect it [8, 66]. The widespread adoption of medical IoT has also introduced new security and privacy questions and concerns. These security and privacy concerns emerge at multiple stages in the life-cycle of the data [58]. In the data transmission for example, [70] proposed a method to capture network traffic from medical IoT devices and automatically detect clear-text information that may reveal sensitive medical conditions and behaviors. [6], in turn, presented PDI, a framework which aims to prevent an adversary from inferring certain sensitive information about subjects using the encrypted data that they disclosed during communication with an intended recipient. Other approaches such as the NeuroSENS architecture [23] tries to improve the security and the privacy of neurogical gait monitoring at several levels (data storage, mobile and web apps and data transmission). Although gesture recognition attracts many attention currently [69], to the best of our knowledge, our work is the first one that addresses the protection of data dedicated to activity recognition through wearable devices in the medical domain. The identification of relevant features for both the activity recognition and the user re-identification is also novel.

Several well known reported user re-identifications have shown that hiding explicit identity information through pseudonymity is not enough to guarantee the anonymity of users [44]. Indeed, many criteria lead to uniquely identifying users. Previous researches have shown that individuals can be identified from their mobility [12, 48], their touch-based gestures on touch-screen devices [49], or their Web browsers [21] to name a few. Following these studies, we also demonstrate in this paper that an user can be easily identified from its gestures collected by sensors.

Compared to other approaches that obfuscate independently every record (e.g., based on differential privacy [7]), only features leading to the re-identification of users are obfuscated. In addition, although this obfuscation based on a normalization does not provide the same privacy guaranty as other generalization-based approaches ensuring k-anonymity, the utility (i.e., activity recognition) remains high while providing a good privacy (i.e., a small re-identification rate). In case of insufficient ground-truth datasets, [30, 71] try to leverage shared information between different classes (i.e., activities with similar patterns in term of signal) to improve the classification.

Lastly, splitting sensitive information (i.e., both the identity of users and their data) on different

nodes have already shown its benefits in terms of privacy [31, 52]. In addition, by processing the signals at the edge of the network on the smartphone of users, our framework inherently reduces the operational costs of the application [13] and strengthens the control of users on their data.

Finally, several initiatives start to leverage homomorphic encryption for machine learning [25, 36, 37]. For instance, [11] describes a method that consists of building blocks of homomorphic functions which they later use to compose several machine learning schemes. Although recent advances in homomorphic encryption schemes improve the performances [17, 19], these solutions are still resource consuming and face to scalability problems. In terms of available libraries, TFHE [3] provides the best performance for binary encoding while HEEAN [2] is the best one for floating point operation support.

8 Conclusion

We present a privacy-preserving IoT framework in the context of activity recognition for healthcare monitoring with wearable devices. Our framework processes the signal and extracts relevant features locally on the user smartphone. In addition, accordingly to the observation that the frequency domain prevails in the user identification task, a normalization is performed on the frequency-based features to obfuscate the re-identification of users. Finally, only a set of features unlinked to the identity of its owner is uploaded to the application server which is then able to recognise the activity of the users with a high accuracy while reducing the risk of user re-identification. An extensive validation of our framework has been performed on 2 reference datasets yielding good results in terms of privacy-utility trade-off and suggesting that the approach could be generalized. However, the different datasets were collected from a smartphone and it would therefore be necessary to evaluate our approach on data recorded via other mobile devices such as objects connected to the smartphone (e.g., smartwatch). Finally, it should be noted that the proposed framework can easily be transposed to other health applications (e.g., calorimetry, pulse, sleep analysis).

References

[1] Amazon elastic compute cloud (amazon ec2). `http://aws.amazon.com/ec2.`

[2] Homomorphic encryption for arithmetic of approximate numbers. https://github.com/snucrypto/HEAAN.

[3] Tfhe: Fast fully homomorphic encryption over the torus. `https://tfhe.github.io/tfhe/`.

[4] G. Acs and C. Castelluccia. A case study: Privacy preserving release of spatio-temporal density in paris. In KDD, pages 1679–1688, 2014.

[5] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In ESANN, 2013.

[6] D. Aranki and R. Bajcsy. Private disclosure of information in health tele-monitoring. CoRR, abs/1504.07313, 2015.

[7] R. Assam, M. Hassani, and T. Seidl. Differential private trajectory obfuscation. In MOBIQUITOUS, pages 139–151, 2013.

[8] E. Ayday and M. Humbert. Inference attacks against kin genomic privacy. S&P, 15(5):29–37, 2017.

[9] C. BenAbdelkader, R. Cutler, and L. Davis. Stride and cadence as a biometric in automatic person identification and verification. In FG, pages 372–377, 2002.

[10] S. D. Bersch, D. Azzi, R. Khusainov, I. E. Achumba, and J. Ries. Sensor data acquisition and processing parameters for human activity classification. Sensors, 14(3):4239–4270, 2014.

[11] R. Bost, R. A. Popa, S. Tu, and S. Goldwasser. Machine learning classification over encrypted data. IACR Cryptology ePrint Archive, 2014:331, 2014.

[12] A. Boutet, S. Ben Mokhtar, and V. Primault. Uniqueness Assessment of Human Mobility on Multi-Sensor Datasets. Research report, LIRIS UMR CNRS 5205, Oct. 2016.

[13] A. Boutet, D. Frey, R. Guerraoui, A.-M. Kermarrec, and R. Patra. Hyrec: Leveraging browsers for scalable recommenders. In Middleware, pages 85–96, 2014.

[14] L. Breiman. Random forests. Machine learning, 45(1):5–32, 2001.

[15] J. Bushberg, J. Seibert, E. Leidholdt, and J. Boone. The Essential Physics of Medical Imaging. Wolters Kluwer Health, 2011.

[16] I. Cheong, S. An, W. Cha, M. Rha, S. Kim, D. Chang, and J. Hwang. Efficacy of mobile health care application and wearable device in improvement of physical performance in colorectal cancer patients undergoing chemotherapy. Clinical Colorectal Cancer, 17(2):e353 – e362, 2018.

[17] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène. Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds. In ASIACRYPT, pages 3–33, 2016.

[18] J. Domingo-Ferrer, D. Sánchez, and J. Soria-Comas. Database Anonymization: Privacy Models, Data Utility, and Microaggregation-based Inter-model Connections. Synthesis Lectures on Information Security, Privacy, & Trust. Morgan & Claypool Publishers, 2016.

[19] L. Ducas and D. Micciancio. Fhew: Bootstrapping homomorphic encryption in less than a second. In EUROCRYPT, pages 617–640, 2015.

[20] C. Dwork. Differential Privacy. In Automata, Languages and Programming, volume 4052, pages 1–12. 2006.

[21] P. Eckersley. How unique is your web browser? In PETS'10, pages 1–18, 2010.

[22] C. Frindel and D. Rousseau. How accurate are smartphone accelerometers to identify intermittent claudication? In HealthyIoT, pages 19–25, 2017.

[23] P. Gard, L. Lalanne, A. Ambourg, D. Rousseau, F. Lesueur, and C. Frindel. A secured smartphone-based architecture for prolonged monitoring of neurological gait. In HealthyIoT, pages 3–9, 2018.

[24] J. Gehrke, E. Lui, and R. Pass. Towards privacy for social networks: A zero-knowledge based definition of privacy. In TCC, pages 432–449, 2011.

[25] C. Gentry, A. Sahai, and B. Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In CRYPTO, pages 75–92, 2013.

[26] R. Geyer, T. Klein, and M. Nabi. Differentially private federated learning: A client level perspective. CoRR, abs/1712.07557, 2017.

[27] O. Goldreich. Cryptography and cryptographic protocols. Distrib. Comput., 16(2-3):177–199, 2003.

[28] M. Gramaglia and M. Fiore. Hiding mobile traffic fingerprints with GLOVE. In CoNEXT, pages 26:1–26:13, 2015.

[29] B. Gregorutti, B. Michel, and P. Saint-Pierre. Correlation and variable importance in random forests. Statistics and Computing, 27(3):659–678, 2017.

[30] T. Gu, L. Wang, H. Chen, X. Tao, and J. Lu. Recognizing multiuser activities using wireless body sensor networks. IEEE Transactions on Mobile Computing, 10(11):1618–1631, Nov 2011.

[31] S. Guha, M. Jain, and V. N. Padmanabhan. Koi: A location-privacy platform for smartphone apps. In NSDI, pages 183–196.

[32] A. Gupta, T. Stewart, N. Bhulani, Y. Dong, Z. Rahimi, K. Crane, C. Rethorst, and M. Beg. Feasibility of wearable physical activity monitors in patients with cancer. JCO Clinical Cancer Informatics, (2):1–10, 2018.

[33] M. Haghi, K. Thurow, and R. Stoll. Wearable devices in medical internet of things: scientific research and commercially available devices. HIR, 23(1):4–15, 2017.

[34] J. Han, J. Pei, and M. Kamber. Data mining: concepts and techniques. Elsevier, 2011.

[35] J. Henriksen-Bulmer and S. Jeary. Re-identification attacksa systematic literature review. International Journal of Information Management, 36(6, Part B):1184 – 1192, 2016.

[36] E. Hesamifard, H. Takabi, and M. Ghasemi. Cryptodl: Deep neural networks over encrypted data. arXiv preprint arXiv:1711.05189, 2017.

[37] E. Hesamifard, H. Takabi, and M. Ghasemi. Deep neural networks classification over encrypted data. CODASPY, pages 97–108, 2019.

[38] G. James, D. Witten, T. Hastie, and R. Tibshirani. An introduction to statistical learning, volume 112. Springer, 2013.

[39] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler. Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. TITB, 10(1):156–167, 2006.

[40] P. Kasnesis, C. Patrikakis, and I. Venieris. Perceptionnet: A deep convolutional neural network for late sensor fusion. Proceedings of the 2018 Intelligent Systems Conference (IntelliSys) Volume 1, pages 101–119, 01 2019.

[41] J. Konecný, H. Brendan McMahan, D. Ramage, and P. Richtárik. Federated optimization: Distributed machine learning for on-device intelligence. CoRR, abs/1610.02527, 2016.

[42] J. Konecný, H. Brendan McMahan, F. X. Yu, P. Richtárik, A. Theertha Suresh, and D. Bacon. Federated learning: Strategies for improving communication efficiency. CoRR, abs/1610.05492, 2016.

[43] D. R. Krishnan, D. L. Quoc, P. Bhatotia, C. Fetzer, and R. Rodrigues. Incapprox: A data analytics system for incremental approximate computing. In WWW, pages 1133–1144, 2016.

[44] L. L. Confidentiality and privacy of electronic medical records. JAMA, 285(24):3075–3076, 2001.

[45] N. Li, T. Li, and S. Venkatasubramanian. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. In ICDE, pages 106–115, 2007.

[46] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. l-diversity: Privacy Beyond k-anonymity. ACM Transactions on Knowledge Discovery from Data, 1(1), 2007.

[47] M. Malekzadeh, R. G. Clegg, A. Cavallaro, and H. Haddadi. Protecting sensory data against sensitive inferences. In W-P2DS, pages 2:1–2:6, 2018.

[48] D. Manousakas, C. Mascolo, A. R. Beresford, D. Chan, and N. Sharma. Quantifying privacy loss of human mobility graph topology. PETS, 2018(3):5–21, 2018.

[49] R. Masood, B. Z. H. Zhao, H. J. Asghar, and M. A. Kâafar. Touch and you're trapp(ck)ed: Quantifying the uniqueness of touch gestures for tracking. PoPETs, 2018(2):122–142, 2018.

[50] S. Mehrang, J. Pietilä, and I. Korhonen. An activity recognition framework deploying the random forest classifier and a single optical heart rate monitoring and triaxial accelerometer wrist-band. Sensors, 18(2):613, 2018.

[51] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. Journal of machine learning research, 12(Oct):2825–2830, 2011.

[52] A. PETIT, T. Cerqueus, S. Ben Mokhtar, L. Brunie, and H. Kosch. PEAS: Private, Efficient and Accurate Web Search. In TrustCom, 2015.

[53] A. Petit, T. Cerqueus, A. Boutet, S. B. Mokhtar, D. Coquil, L. Brunie, and H. Kosch. Simattack: private web search under fire. Journal of Internet Services and Applications, 7(1):1–17, 2016.

[54] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai. Privacy-preserving deep learning via additively homomorphic encryption. IEEE Transactions on Information Forensics and Security, 13(5):1333–1345, 2018.

[55] I. M. Pires, N. M. Garcia, N. Pombo, and F. Flórez-Revuelta. From data acquisition to data fusion: a comprehensive review and a roadmap for the identification of activities of daily living using mobile devices. Sensors, 16(2):184, 2016.

[56] S. J. Preece, J. Y. Goulermas, L. P. Kenney, D. Howard, K. Meijer, and R. Crompton. Activity identification using body-mounted sensorsa review of classification techniques. Physiological measurement, 30(4):R1, 2009.

[57] J. L. Reyes-Ortiz. Smartphone-based human activity recognition. Springer, 2015.

[58] M. Rushanan, A. D. Rubin, D. F. Kune, and C. M. Swanson. Sok: Security and privacy in implantable medical devices and body area networks. In S&P, pages 524–539, 2014.

[59] S. Scalvini, D. Baratti, G. Assoni, M. Zanardini, L. Comini, and P. Bernocchi. Information and communication technology in chronic diseases: a patients opportunity, 2014.

[60] J. Schrack, G. Gresham, and A. Wanigatunga. Understanding physical activity in cancer patients and survivors: New methodology, new challenges, and new opportunities. Molecular Case Studies, 3:mcs.a001933, 04 2017.

[61] B. Seref and E. Bostanci. Opportunities, threats and future directions in big data for medical wearables. In BDAW, pages 15:1–15:5, 2016.

[62] M. Shoaib, S. Bosch, O. Incel, H. Scholten, and P. Havinga. Fusion of smartphone motion sensors for physical activity recognition. Sensors, 14(6):1014610176, 2014.

[63] S. Sprager and M. B. Juric. Inertial sensor-based gait recognition: a review. Sensors, 15(9):22089–22127, 2015.

[64] L. Sweeney. k-Anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(5):557–570, 2002.

[65] Y. Tang and C. Ono. Detecting activities of daily living from low frequency power consumption data. In MOBIQUITOUS, pages 38–46, 2016.

[66] F. Tramèr, Z. Huang, J.-P. Hubaux, and E. Ayday. Differential privacy with bounded priors: Reconciling utility and privacy in genome-wide association studies. In CCS, pages 1286–1297, 2015.

[67] J. Wang, L. Cadmus-Bertram, L. Natarajan, M. White, H. Madanat, J. Nichols, G. Ayala, and J. Pierce. Wearable sensor/device (fitbit one) and sms text-messaging prompts to increase physical activity in overweight and obese adults: A randomized controlled trial. Telemedicine and e-Health, 21(10):782–792, 2015.

[68] Y. Wang, X. Wu, and D. Hu. Using randomized response for differential privacy preserving data collection. In EDBT, 2016.

[69] H. Watanabe, T. Terada, and M. Tsukamoto. Gesture recognition method based on ultrasound propagation in body. In MOBIQUITOUS, pages 288–289, 2016.

[70] D. Wood, N. Apthorpe, and N. Feamster. Cleartext data transmissions in consumer IoT medical devices. In IoT S&P, pages 7–12, 2017.

[71] L. Yao, F. Nie, Q. Z. Sheng, T. Gu, X. Li, and S. Wang. Learning from less for better: Semi-supervised activity recognition via shared structure discovery. In Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pages 13–24, 2016.

[72] Z.-K. Zhang, M. C. Y. Cho, C.-W. Wang, C.-W. Hsu, C.-K. Chen, and S. Shieh. Iot security: ongoing challenges and research opportunities. In SOCA, pages 230–234, 2014.