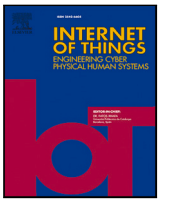
Internet of Things 26 (2024) 101189

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot



Research article

Privacy preservation in sensor-based Human Activity Recognition through autoencoders for low-power IoT devices

Leonardo Bigelli, Chiara Contoli, Valerio Freschi, Emanuele Lattanzi *

Department of Pure and Applied Sciences, University of Urbino, Piazza della Repubblica 13, Urbino, 61029, Italy

ARTICLE INFO

Keywords:
Privacy preservation
Sensor-based HAR
Deep-learning
Low-power IoT devices

ABSTRACT

Human activity recognition is increasingly recognized as a key task in many applications. However, gathering data from the variety of sensors commonly available on end devices risks compromising user's privacy when signals are transmitted to more powerful computing units for inference offloading. It is therefore important to design and implement strategies that could prevent privacy breaches without impairing the capability of the system of recognizing activity patterns, and by taking into account the energy constraints of low-power devices. In this work, we propose an energy-aware approach aimed at preserving the privacy of users during inference of human activities. The proposed method is based on a deep learning autoencoder trained to process the signal in order to remove the most sensitive information regarding privacy attributes, without significantly impacting classification accuracy. We also perform a thorough architecture's parameter tuning of the designed system to enable its implementation on a low-power platform, which we also characterize in terms of energy expenditure. Experimental results show that this system is capable of effectively transforming the signal in order to prevent the inference of sensitive attributes (i.e. weight, height, age, and gender) and it can be conveniently implemented on a constrained embedded system at different levels of the trade-off between accuracy and energy consumption. Indeed, a complete obfuscation of sensitive attributes can be achieved at the cost of a marginal reduction in classification accuracy (5% at most), with an expenditure of around 165 mJ for an execution time of around 30ms needed during the signal transformation step.

1. Introduction

Due to significant technological advancements in system-on-chip (SoC) design, the growth of connected wearable devices has experienced remarkable strides in recent years. According to the wearable technology market report, the market size for wearable devices reached USD 71.91 billion in 2023, with the global number of connected wearables surpassing 1.1 billion [1,2]. Wearable devices encompass a diverse range of products such as smartwatches, smartbands, chest straps, shoes, helmets, glasses, lenses, rings, patches, textiles, and hearing aids. Of these, smartwatches and smartbands cover the vast majority of the current market [3] . From a hardware perspective, these devices vary from the sophisticated 64-bit multiple-core processors found in contemporary smartwatches to the energy-efficient 16 or 32-bit constrained microcontrollers present in smartbands, shoes, rings, and similar devices.

The widespread adoption of wearable Internet of Things (IoT) devices has led to the collection of vast amounts of data, enabling innovative applications in various domains [4]. Human Activity Recognition (HAR) has garnered significant attention among these applications due to its potential in healthcare monitoring and elderly care [5], and smart homes [6]. In 2014, MIT defined "the Era

* Corresponding author.
E-mail address: emanuele.lattanzi@uniurb.it (E. Lattanzi).

https://doi.org/10.1016/j.iot.2024.101189
Received 23 November 2023; Received in revised form 8 April 2024; Accepted 15 April 2024
Available online 16 April 2024

of Ubiquitous Listening" [7] with the advent of voice assistants, which are known to suffer from privacy risks coming mainly from two causes: (1) raw audio signals are shared with the cloud of service providers, which might raise potential inference attacks, and (2) the fact that those signals contain sensitive paralinguistic information [8]. Indeed, deep acoustic models can use paralinguistic information to infer personal and sensitive data such as speaker identity, gender, age, mood, emotions, ethnicity, and health status, to cite a few. In the HAR domain, to use the parallelism with "the Era of Ubiquitous Listening", we can say that we are entering "the Era of Ubiquitous Motion Tracking". The abundance of personal data collected by these devices that incorporate multiple sensors (such as accelerometers, gyroscopes, and magnetometers) raises serious concerns regarding individual privacy. For instance, these devices continuously transmit collected data to a remote server to identify user activities, such as walking, jogging, and running. However, individuals perform these activities differently based on personal factors like age, gender, and weight. For example, a person with a higher body weight may walk slower than someone with a lower body weight. Hence, interpreting sensor signals allows for the determination of personal information like weight, age, gender, and height without the user's conscious participation or explicit permission, thus becoming a potential threat despite being generally thought to be harmless [9–11].

To safeguard privacy, the direct transmission of sensor data without privacy protection measures is unacceptable. However, it is vital to maintain activity recognition accuracy so as not to make the application unusable. Recently, Antwi-Boasiako et al. surveyed privacy preservation techniques in distributed deep learning [12]. In these scenarios, privacy issues can arise at several stages, such as during dataset creation, feature extraction, and model training. The authors in that study focus on privacy issues at the training stage, although they also highlight some privacy concerns that may appear at the inference stage. As privacy preservation methods the authors mention: (i) differential privacy (DP), (ii) homomorphic encryption (HE), and (iii) secure multi-party computations (SMPC). However, it is worth mentioning that other strategies have been employed for privacy preservation purposes, such as for example generative adversarial networks (GANs) and autoencoders. All these methods might be grouped into transformation and perturbation methods.

The adoption of autoencoders seems a promising approach for privacy enhancement in different application domains such as speech recognition [8] and face de-identification [13]. Autoencoders are neural networks that are designed to compress (encode) the input into a telling representation, and in a second phase the input is decoded in such a way that the outcome is similar as much as possible to the original input. In [14], the authors survey the most widely adopted autoencoders, highlighting in particular the so called regularization techniques, i.e., the way by which autoencoders ensure that the compressed representation of the input is meaningful. It is worth mentioning that autoencoders can be used in combination with other methods, for example, GANs [15], in order to take advantage of both methodologies.

Applying autoencoders for privacy enhancement in signals from inertial sensors is not new: many works in the literature focus on proposing data transformation mechanisms to anonymize sensor data [16–18], others apply such mechanisms to anonymize one specific sensitive data (e.g., gender [19]). Although the blooming literature, those works do not consider the design of a multiple attributes obfuscator (i.e., an autoencoder capable of anonymizing multiple sensitive information at a time) nor the deployment of those systems on real low-power constrained devices, thus overlooking the energy consumption obfuscation time factors.

In this article, we propose an energy-aware privacy preservation strategy dedicated to low-power constrained devices and based on a deep-learning autoencoder trained to disguise users' sensitive information while maintaining high accuracy in activity recognition. In particular, the contribution of this work can be summarized as follows:

- we propose an approach for preventing unwanted inference of *multiple* privacy attributes (i.e., gender, height, weight, age) from signals acquired by means of inertial sensors;
- we provide a thorough parameter tuning investigation of the model architecture and the proposed methodology;
- we implement the autoencoder-based obfuscation on a real low-power platform, demonstrating the feasibility of porting this architecture on severely constrained systems;
- we characterize, through an extensive set of experiments, the effectiveness and efficiency of our approach along multiple dimensions: (i) accuracy of the HAR models; (ii) privacy preservation capabilities; (iii) energy consumption;

The remainder of the article is organized as follows: in Section 2, we report the state-of-the-art research related to our study; in Section 3, we introduce the learning architecture and system design choices proposed as a solution; in Section 4, we illustrate the experimental setup adopted for performance assessment; in Section 5, we describe and discuss the experimental results; in Section 6, we conclude by summarizing the key findings of our investigation.

2. Related work

Privacy preservation is about avoiding the leakage of personal and sensitive information. Human Activity Recognition systems utilize sensor data from low-power wearable IoT devices to constantly infer what kind of activity we are doing. Therefore, individual privacy become a severe concern. In this paper, we do not consider work related to privacy preservation in vision-based activity recognition. Instead, we focus on privacy preservation in inertial sensor-based activity recognition.

Privacy Preserving in Inertial Sensor-based Activity Recognition: Many privacy-preserving strategies have been investigated by the research community. Lyu et al. proposed a two-stage scheme based on randomization/perturbation techniques for privacy preservation improvement in collaborative deep learning [20]. They implement a two-stage approach: in the first, they apply a nonlinear function to perturb data; in the second, they compress the data via a row-orthogonal random projection matrix to preserve the Euclidean distance between pairs of data points stochastically. This approach shows promising results in terms of good trade-off between privacy and accuracy recognition.

Zhang et al. propose a framework that combines both perturbation noise and transformation methods: it transforms raw sensor data into a new representation with a "style" of random noise, i.e., sensitive information, and a "content" of raw data, i.e., activity information [21]. They use a 2D convolutional neural network to train raw sensor data to infer target information and then train the transformation network to avoid information leakage while keeping the target information. The framework simultaneously protects user-sensitive information and retains target information with a low recognition accuracy drop. Garain et al. also propose a HAR framework such that the user data are differentially private [22]. They perform uniform noise distribution by adding noise to the original data via the linear shifting mechanism to achieve this. They measure the privacy and recognition performances using various levels of noise insertion. However, they use a multilayer perceptron instead of deep learning as a classifier and consider only the user identifier as sensitive information. They show that the HAR accuracy remains high when noise is increasingly inserted into accelerometer data, whereas the user identifier accuracy drops significantly.

Climent-Pérez and Florez-Revuelta leverage the nondominated sorting genetic algorithm III to minimize the identification of gender and age recognition while maximizing the activity recognition [23]. The evolutionary algorithm's main objective is to find the best features to perform HAR as accurately as possible while concealing gender and age. Despite gender and age recognition experience a significant drop in accuracy, there can be a non-negligible drop in activity recognition because of the multidimensionality of the problem. Indeed, the solution could be optimal for some dimensions but suboptimal for others.

Menasria et al. propose two frameworks based on two different adversarial training models against private information inference [24]. The first approach, private GAN1, is based on a dual discriminator architecture. PGAN1 transforms, i.e., disguises (raw data) sensitive features related to private attributes while keeping non-sensitive features related to non-sensitive attributes. The second approach, PGAN2, leverages random distribution instead of transformation to generate new data with a minimum correlation to the distribution of sensitive features. PGAN2 is based on one discriminator and two estimators architecture. The authors tested the two strategies by anonymizing one or more sensitive information at a time (e.g., identity, gender, age, weight, height); they concluded that PGAN2 is more effective than PGAN1 in concealing sensitive features.

Malekzadeh et al. developed a privacy-preserving platform based on autoencoders to disguise sensitive information in time-series data analysis [25]. The basic idea is to introduce a deep autoencoder architecture on which a real-time algorithm can extract useful features from time-series data; this algorithm then learns how to replace sensitive inferences with non-sensitive ones, whereas desired information is retained in data. This approach is called replacement autoencoder (RAE). The RAE system is, therefore, trained to learn how to transform discriminative features corresponding to sensitive inferences into features observed in non-sensitive inferences. To prove the effectiveness of their approach, the authors used a Generative Adversarial Network (GAN), showing that the output produced by the RAE is indistinguishable from the real non-sensitive information. They also prove that RAE can simultaneously retain the recognition accuracy.

Subsequently, Malekzadeh et al. extended their previous work by proposing the Guardian Estimator Neutralizer (GEN) framework, which is designed to share only a transformed version of the sensor data [26]. The guardian component is a learning framework based on the deep autoencoder described in their previous work mentioned above; the estimator is a multi-task convolutional neural network in charge of quantifying how accurate an algorithm can make sensitive and non-sensitive inferences on the transformed data. The neutralizer is an optimizer that helps the autoencoder, i.e., the guardian, to find the optimal transformation function to infer an activity without revealing sensitive information. In this case, the authors consider only the user's gender as sensitive information. They proved that the GEN provides a good compromise between utility and accuracy.

An approach similar to content and style was also proposed by Delgado et al. designing GaitPrivacyON [27]. GaitPrivacyON consists of two modules. The first one includes two convolutional autoencoders that share weights and have the same architecture; these autoencoders are trained to extract helpful information from transformed data. Sensitive data are instead disguised by applying random noise. The second module is the gait verification system, which is composed of a convolutional neural network combined with a recurrent neural network that helps the system keep its usefulness in the main task of gait verification. They obtained good results because the gait biometrics verification task slightly decreased. In contrast, sensitive data protection, in this case, activity and gender, increased.

Deep Autoencoder Application and Deployment: Deep autoencoders are applied in many fields of applications and with different purposes. The reason for this success is twofold: on the one hand, deep autoencoder serves several application domains and goals; on the other one, although being a deep neural network, the research community has proven the feasibility of its deployment on low-power devices. One relevant goal of deep autoencoders is anomaly detection, which finds application, for example, in the industrial environment and health care monitoring. Successful examples of anomaly detection via autoencoders in the industrial environment are: acoustic/vibration anomaly detection for predictive maintenance purposes deployed on an ARM Cortex/Arduino, and the ESP32 devices [28–31]; identification of malicious network traffic in the IoT ecosystem, despite being only simulated and not physically deployed [32]. Other successful examples of anomaly detection are in the smart city field [33] and in the healthcare monitoring field: normal or abnormal detection of electrocardiography data for the diagnosis of diseases, deployed on the ESP32 [34]; encryption and abnormality detection of photoplethysmogram signals and temperature to estimate the heart rate, blood oxygen saturation, respiration rate, and blood pressure, to secure vital data, deployed on the ESP32 [35].

Contributions: Our work is inspired by those of Malekzadeh et al. [25,26], i.e., we leverage the idea of adopting an autoencoder that retains aspects of the original input that are useful and relevant to the activity recognition, while masking sensitive data that might incur privacy leakages. However, compared to their work and the existing literature, we hide all the sensitive information (gender, weight, height, and age) simultaneously, thus creating a multiple attributes obfuscator. Moreover, we also carry out the energy characterization of this privacy-preserving method that, to the best of our knowledge, was neglected in the existing literature.

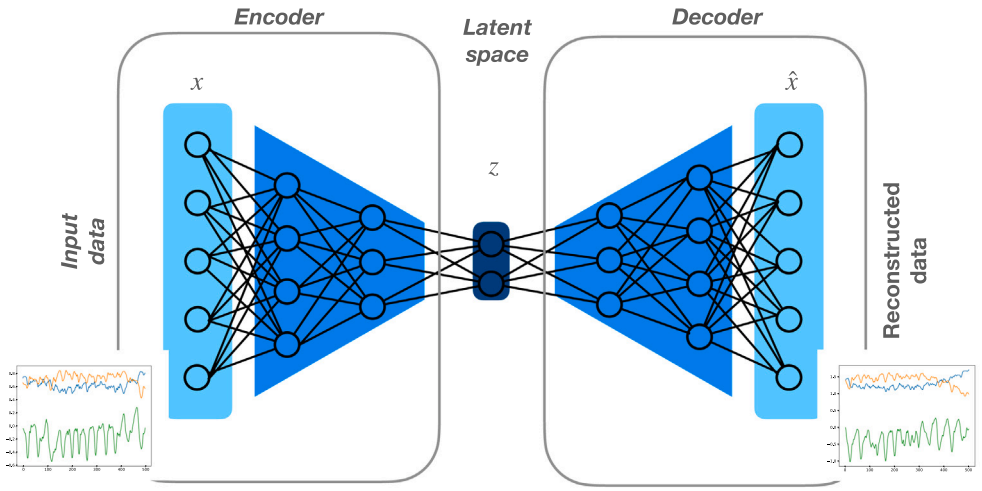


Fig. 1. Autoencoder general architecture.

3. Proposed approach

In this section, we describe the proposed methodology for privacy-preserving HAR. The problem can be formally stated as follows: we are given a set of N_s synchronized sensors producing (at the same rate) a signal $S(t) = [s_1(t), s_2(t), \ldots, s_{N_s}(t)]$. Given a time window of length w, we consider a portion of signal $S_w(t) = [S(t), S(t+1), \ldots, S(t+w-1)]$. $S_w(t)$ (hereafter also denoted as S_w for simplicity) is a matrix representing the values gathered by sensors during a time window of length w started at time t. In a typical application scenario, data produced by the mobile user device are transmitted remotely to a cloud-based service provider to infer the corresponding activities (through machine learning models), provide feedback to users, and possibly store data. In this setting, the threat model we assume is that of a "honest but curious" adversary model in which the service provider, which has legitimate access to the user's inertial signals, can (possibly) try to infer additional sensitive information from sensor streams (e.g. gender, age, or physical attributes) compromising user privacy.

Given the knowledge of the attacker (i.e., the service provider) machine learning model, the proposed approach aims to keep the capability of correctly recognizing activities (utility) to adequate levels while protecting from potential unauthorized inference of sensitive features (privacy). To achieve this goal, we propose a system based on a deep learning autoencoder trained to process input data (i.e., the sensor stream S_w) into a transformed version ($\hat{S}_w$) that retains at most the information useful for classification of nonsensitive elements. At the same time, it makes inference of sensitive data difficult. Given the HAR classifier model, the autoencoder is trained using a specific loss function composed of two terms, one in charge of incentivizing correct inference of legitimate attributes, the other promoting the penalization of the accurate prediction of different types of sensitive information.

In particular, we aim to take advantage of the capability of autoencoders (based on neural networks) to learn a representation of the input data through a non-linear projection on a latent space and its subsequent reconstruction.

Privacy protection in wearable and mobile computing systems represents a broad topic with multiple issues that must be carefully considered at design time, specifically concerning the balance between privacy and utility. Lin et al. discussed the concept of resilience as a way to provide guarantees in terms of robustness and adaptability [36]; this concept can be leveraged to complement privacy protection. Indeed, in the case of our proposed privacy-preserving HAR system, privacy and (more in general) security risks could be related to different types of attacks. For instance, the autoencoder could be bypassed to directly access the inertial signals, clearly making the proposed strategy ineffective; another possible vulnerability could be represented by illegitimate access (e.g., to the disk memory of the device) to the structure and weights of the autoencoder, which could potentially be exploited for reverse-engineering the obfuscation system and compromise its functionality. Resilience with respect to these types of attacks is typically enforced through the devices' operating system. Other strategies to support resilience could leverage authentication mechanisms based on symmetric key generation systems as discussed in [36–38], which was revealed to be helpful also in case of eavesdropping attacks.

In principle, other potential attacks and privacy risks could be envisioned. However, those are beyond the scope of this article and might be considered as future work. Here, we focus on safeguarding sensitive attributes of inertial sensor data.

Autoencoders are neural networks designed to compress (encode) the input into a telling representation, and, in a second phase, this input is decoded so that the outcome is similar as much as possible to the original input. In [14], Bank et al. survey the most widely adopted autoencoders, highlighting, in particular, the so-called regularization techniques, i.e., how autoencoders ensure that the compressed representation of the input is meaningful.

Fig. 1 shows a generic architecture of a deep learning autoencoder. Since the encoding-decoding pipeline allows the network to produce an approximate copy of the input, by properly prioritizing which aspects of the input should be replicated, it is, in principle, possible to produce data that retains some interesting aspects of the original input (e.g., helpful for activity recognition), while obfuscating others that could impair privacy.

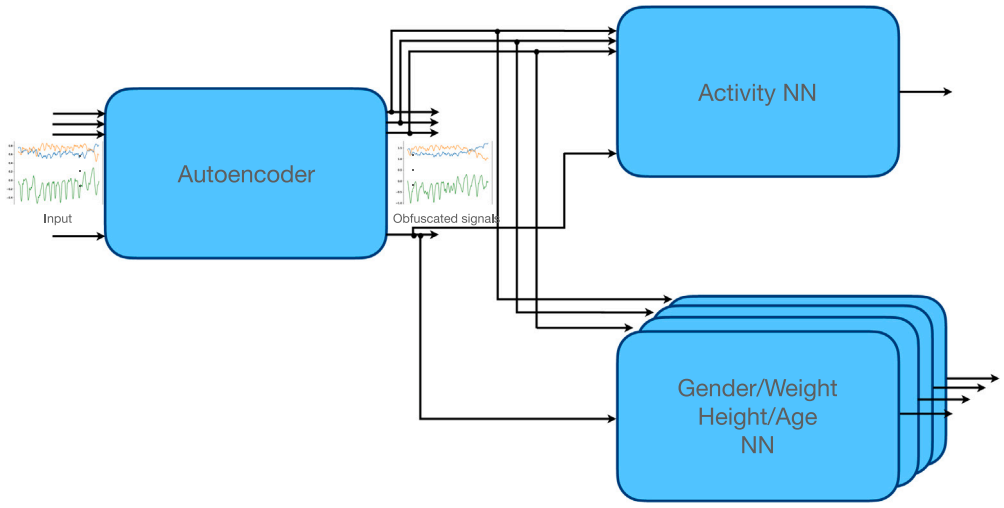


Fig. 2. Overall system architecture.

3.1. Model architecture

To design the system, we follow the approach introduced by Malekzadeh et al. in [26], where an autoencoder performs a transformation of a given window segment of multidimensional time series into an output of the same dimensionality. The transformed version of the time series segment can be subsequently fed into an activity recognition module. At the same time, if it is given as input into specific neural networks for performing inference of sensitive attributes, it should not reveal any (or few) attributes, thus preserving privacy without significantly reducing the accuracy of legitimate activity recognition.

To that aim, our proposal entails the design of: (i) a deep learning model for the inference of non-sensitive information (i.e., for standard activity classification); (ii) a set of deep learning models (four in our case study) capable of making inference of sensitive information (one for each sensitive attribute, i.e., for gender, weight, height, and age), as depicted in Fig. 2.

The autoencoder is indeed trained according to a feedback loop scheme, which resembles adversarial training: the weights of the activity classifier and the sensitive-information inference models are frozen (declared no more trainable). Then, by using a specifically designed loss function $\mathcal{L}$, which rewards correct recognition of non-sensitive (from the privacy point of view) activities while penalizing accurate estimates of sensitive attributes, the weights of the autoencoder are learned. In particular, the loss function can be expressed as the sum of two contributions:

$$\mathcal{L} = \mathcal{L}_a + \mathcal{L}_s \tag{1}$$

Where the first term, $\mathcal{L}_a$ is the standard categorical cross-entropy, namely:

$$\mathcal{L}_a = -\sum_{i=1}^{C} y_a^i(S_w) \log \hat{y}_a^i(\hat{S}_w) \tag{2}$$

In Eq. (2), S_w represents the raw signal, i.e., the time series stream acquired from sensors during a time window of length w, $\hat{S}_w$ is the transformation of the signal according to the autoencoder; $y_a^i(S_w)$ is the true activity label of the sample S_w; $\hat{y}_a^i(\hat{S}_w)$ is the predicted value for the related class of activity, and C is the number of classes.

The second term, $\mathcal{L}_s$,is computed differently if we consider regression or classification as a privacy-sensitive inference task.

Regression:

In the case of regression (for instance, the task is the inference of weight, height, or age of a person), we formulated $\mathcal{L}_s$ as a quadratic curve independent from the true value to be predicted:

$$\mathcal{L}_s = 0.1 \cdot [y_{max} - \hat{y}_s(\hat{S}(w))]^2 \tag{3}$$

In Eq. (3), $\hat{y}_s(\hat{S}(w))$ represents the regression value predicted by the model, while y_{max} represents a value twice the maximum that the quantity to be predicted could hypothetically take; for instance, for age prediction, assuming 100y as maximum value, $y_{max} = 200$, while $y_{max} = 2 * 100$ for weight (100 kg max), and $y_{max} = 2 * 200$ (200 cm max) in the case of height.

Classification:

In the case of classification (for example if the task is the inference of gender), $\mathcal{L}_s$ is computed as:

$$\mathcal{L}_s = |\frac{1}{2} - y_s(\hat{S}(w))| \tag{4}$$

$y_s(\hat{S}(w))$ is the predicted class, which is the sensitive information (i.e., the gender) that we do not want to be correctly inferred from the sample; hence, the introduction of the value 0.5, which represents the desired confidence for a classifier that performs inference on the transformed data.

Overall, both $\mathcal{L}_a$ and $\mathcal{L}_s$ contribute to shaping the loss function, which is used to train the autoencoder in a way that transformed signals are substantially too much noisy for performing accurate privacy-sensitive inference, but remain informative for recognition of non-sensitive attributes.

We remark that Eq. (1) can be used to build a series of *single attribute* obfuscators, one for each of the sensitive attributes under study, namely gender, age, weight, and height. Alternatively, we also evaluated the possibility of building a *multiple attributes* obfuscator, i.e., we designed and trained an autoencoder intending to transform the original signal into a version of itself robust concerning leakage of all the privacy-related features. Such an alternative has been done by summing, in Eq. (1), four terms that contribute to the loss $\mathcal{L}_s$ (one for each sensitive feature); conversely, for the case of a single attribute obfuscator, only one of the possible term is taken into consideration for $\mathcal{L}_s$.

3.2. System deployment

A possible way to avoid privacy leaks involves keeping data locally and executing any inference activity directly on the board of the user's device. Such an approach contrasts with the computational requirements needed by many current state-of-the-art deep learning models. Indeed, hardware resources available on constrained devices, especially those based on low-power, low-cost microprocessor units, often hinder the execution of inference tasks based on neural networks. Therefore, specific solutions are needed to compress machine learning models and fit them with the memory and compute resources of a given embedded device [39,40].

Since the autoencoder is a deep-learning neural network, its deployment on severely constrained devices presents the same challenges. To address this issue, we first explored the design space of the model architecture that could comply with the functional requirements (to support privacy-preserving HAR); subsequently, we explored compression strategies to meet the resource needs of a typical embedded system based on MCU. In particular, we implemented the autoencoder model in TensorFlow [41] and then performed model optimization (dynamic and full integer 8-bit quantization) using TensorFlow Lite [42] to compress and deploy it onto an ESP32 device [43].

To summarize, the system architecture consists of three deep learning models:

1. An autoencoder, trained using the above-described loss function, acts as a preprocessing tool to prevent specific privacy leaks that could occur when data is transmitted remotely for classification. The autoencoder is located on thr user's device board and it is based on a fully connected architecture where the number of neurons per layer is progressively reduced from an initial value (in the first layer) to a final one in the encoder (up to the latent space), and then expanded symmetrically in the decoder toward the output. In particular, we tested a given range of models, corresponding to as many different configurations of the architecture, as specified in Table 1.
2. A HAR model based on convolutional networks in charge of performing (legitimate) activity recognition; this model is usually hosted on edge devices or directly on more powerful cloud computing resources. We adopted an architecture based on four 1d convolutional layers followed by an average pooling layer and four dense layers (each interspersed with a dropout layer to prevent overfitting).
3. A set of models capable of carrying out some specific inference of sensitive features that we can hypothesize to be placed anywhere along the communication chain that goes from the user up to the cloud. In the context of privacy protection, we can imagine these models as tools used for the malicious acquisition of sensitive information, which, from an architectural point of view, will implement the state-of-the-art in HAR classification and regression. In particular, a neural network based on four convolutional layers followed by four dense layers has been trained for gender classification. Three networks composed of a cascade of four convolutional layers, an average pooling layer, and six dense layers have been designed for the three regression tasks, namely the inference of weight, age, and height. Fully connected layers have been alternated with dropout layers for these models.

The details of the architectures are specified in Table 1, where we reported, under the column #elements, the number of neurons for the case of fully connected layers; for the autoencoder, we detailed all the parameters tested, corresponding to the number of neurons in each layer, in brackets. In the case of a convolutional layer, under the #elements column, we reported a pair of numbers between brackets, where the first one represents the number of filters and the second one the kernel size. Notice that we make use of the Hyperband Tuner provided by Keras [44,45] to find the best hyperparameters setting of the proposed models except for the autoencoder, for which we provided a manual sweep to characterize the trade-off between privacy preservation and energy consumption.

4. Experimental setup

In this section, we present the experimental setup provided to evaluate the effectiveness and the energy efficiency of the proposed approach.

Table 1
Models architecture summary.

Model	Layer	#elements	Model	Layer	#elements
Autoencoder	fully_conn_1	300	HAR	conv_1	(32, 2)
	fully_conn_2	(150, 75, 50, 37, 30, 25, 21, 18)		conv_2	(224, 4)
	fully_conn_3	(100, 50, 33, 25, 20, 16, 14, 12)		conv_3	(160, 2)
	fully_conn_4	(75, 37, 25, 18, 15, 12, 10, 9)		conv_4	(128, 3)
	fully_conn_5	(100, 50, 33, 25, 20, 16, 14, 12)		fully_conn_1	544
	fully_conn_6	(150, 75, 50, 37, 30, 25, 21, 18)		fully_conn_2	1024
	fully_conn_7	300		fully_conn_3	96
				fully conn_4	#classes
Gender_recognition	conv_1	(160, 2)	Height_recognition	conv_1	(64, 5)
	conv_2	(160, 5)		conv_2	(192, 5)
	conv_3	(256, 5)		conv_3	(224, 2)
	conv_4	(32, 4)		conv_4	(64, 1)
	fully_conn_1	288		fully_conn_1	704
	fully_conn_2	768		fully_conn_2	320
	fully_conn_3	160		fully_conn_3	512
	fully_conn_4	#classes		fully_conn_4	320
				fully_conn_5	992
				fully_conn_6	1
Weight_recognition	conv_1	(64, 1)	Age_recognition	conv_1	(192, 3)
	conv_2	(192, 5)		conv_2	(224, 1)
	conv_3	(64, 5)		conv_3	(96, 5)
	conv_4	(224, 4)		conv_4	(160, 4)
	fully conn_1	128		fully_conn_1	960
	fully conn_2	224		fully_conn_2	576
	fully conn_3	928		fully_conn_3	640
	fully conn_4	896		fully_conn_4	960
	fully conn_5	928		fully_conn_5	896
	fully conn_6	1		fully_conn_6	1

4.1. Datasets

Each experiment was conducted using two publicly available datasets: MotionSense [46] and MobiAct [47].

MotionSense includes data collected from an iPhone 6s in the participant's front pocket. Together with raw native sensors such as the triaxial accelerometer and gyroscope, it also contains data from virtual sensors such as gravity and attitude (pitch, roll, yaw), which are derived from software elaboration and are commonly provided by modern smartphones. Since this work is focused on low-performance IoT devices that would not be able to compute virtual sensors, we considered only the data coming from the native sensors (i.e., accelerometer and gyroscope). In particular, the dataset has been built around a total of 24 participants in a range of gender (10 females and 14 males), age (average 29, min 18, and max 35), weight (average 72 kg, min 40 kg, and max 100 kg) height (average 174 cm, min 161 cm, and max 190 cm). These performed six activities in 15 trials in the same environment and conditions: *downstairs*, *upstairs*, *walking*, *jogging*, *sitting*, and *standing*. To train and test our models, we divide each trial into segments using a sliding window of 50 points, corresponding to 1 s, with 75% Choosing the size of the time window and the percentage of overlap is a non-trivial task because the length of the time window impacts the classification performance of the models [48]. For instance, in HAR tasks, different window lengths have been used in the literature, from 1 s up to 30 s [39,49–51]. In this study, we opted for a 1-second time window and 75% of overlap as Zhang et al. proposed in 2021 to compare with their work directly [21].

MobiAct dataset consists of data collected from the accelerometer, gyroscope, and orientation sensors of a Samsung Galaxy S3 smartphone. This data was gathered from 57 individuals while they performed nine distinct activities. Notably, this dataset aims to replicate real-life activities by having participants carry the smartphone in a loose pocket with random orientation. The orientation sensor relies on software and derives information from the accelerometer and the geomagnetic field sensor. As in the case of MotionSense, we do not consider software sensors due to the low computing capabilities of our target devices.

In order to compare the performance of our approach with Zhang et al. 2021 study [21], we have chosen to work with a subset of the original dataset consisting of data gathered from 44 subjects (14 females and 30 males) engaged in the following four activities: *downstairs*, *upstairs*, *walking*, and *jogging*. The subjects show an average age of 25 years (min 20 and max 47), an average weight of 77 kg (min 50 kg and max 120 kg), and an average height of 176 cm (min 158 cm and max 193 cm).

As for the MotionSense dataset, we divide each trial into segments using a sliding window of length 50 points with 75% of overlap.

4.2. Performance metrics

To evaluate the performance in the classification of multi-class problems, the following quantities have been evaluated for each of the classes of the datasets ($i \in [1 \cdots N]$ is an index that identifies a specific class): TP_i, the number of true positives predicted

for class i; TN_i, the number of true negatives predicted for class i; FP_i, the number of false positives predicted for class i; FN_i, the number of false negatives predicted for class i.

Subsequently, these indicators have been used to compute the following metrics (corresponding to the so-called *macro-averaging* measures) [52]:

$$Precision = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FP_i} \tag{5}$$

$$Recall = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FN_i} \tag{6}$$

$$F1score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{7}$$

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \tag{8}$$

Finally, to evaluate the performance of the models acting on continuous information (i.e., height, weight, and age), we use the mean absolute error (MAE) as the evaluation metric since it is more sensible for minor errors. Therefore, it is more suitable to show the worst-case scenario performance.

4.3. Privacy evaluation

To evaluate the capability of obfuscating sensitive information, we measure the recognition performances of the malicious models before and after the transformation applied through the autoencoder. In particular, if the drop in the recognition capability is negligible, the proposed approach is seen as unsuccessful in protecting user-sensitive information. On the contrary, a high drop certifies the suitability of the proposed method.

In parallel, we also have to consider the possible performance degradation in inferring the desired information. For instance, if the accuracy of inferring human activities experiences a minor decrease after data transformation, we can consider the proposed framework successful. Conversely, if the accuracy notably decreases beyond this threshold, we regard the framework as failing to maintain the desired information.

Zhang et al. [21] to quantify the degree of obfuscation make use of the relative result Δ_r defined as:

$$\Delta_r = \frac{R(After) - R(Before)}{R(Before)} \tag{9}$$

where $R(After)$ and $R(Before)$ are absolute results of transformed and raw data, respectively. To make a direct comparison possible, we adopt the metric proposed by Zhang et al. and, in addition, we introduce a new measure which we call *Random Error Ratio* (RER). RER is the ratio between the classification error and the error a random classifier would make. In particular, it quantifies how much the inference results can be considered close to a random guess or if they are due to a sensible recognition. For instance, if the value of RER is close to one, we can consider the inference results as random, which means, in the case of the recognition of sensitive information, that the obfuscation procedure was successful in keeping the information hidden. On the other hand, low values of RER demonstrate a poor obfuscation capability.

As the two datasets can be considered sufficiently balanced, we have computed the random guess error for the classification problems as $1 - (1/k)$ where k is the number of classes. Otherwise, for regressions, we empirically estimate the random MAE using two different random generators from which we extract 10,000 records in the range of the corresponding attribute. A generator carries out the inference, and the other represents the ground truth. For instance, considering the age attribute, we take [18, 100] years as the reference range, which produces a random MAE of about 27.4y. In this case, if the MAE calculated after the obfuscation equals or exceeds the random MAE, we can argue that the proposed framework manages to hide sensitive information completely.

4.4. IoT device

The IoT device, chosen as representative of the wearable low-power connected devices, was an Espressif ESP32-wroom-32 DevKit [43] connected to an MPU6050 integrated 6-axis motion tracking device that combines a 3-axis gyroscope and 3-axis accelerometer [53]. On top of the hardware device, we developed a real-time application that collects sensor data and transforms it using the autoencoder previously described; in the meantime, we measured the energy consumption of the device to characterize the efficiency of the autoencoder. The application was written in C++ using the ESP-IDF platform (Espressif IoT Development Framework) with the TensorFlow Lite Micro libraries installed. The ESP32 has two CPU cores that can be individually controlled, running at a clock frequency adjustable from 80 MHz to 240 MHz, so the sensing and transforming application is composed of two different tasks, each of which executes on a separate CPU core. One thread continuously collects data from the sensors, acting as a data producer, whereas the second thread behaves like a data consumer by executing the ML model to remove sensitive information from data.

The used version of the ESP32 has 4 MB of flash memory for saving the firmware and 400 KB of RAM. Since a TensorFlow Lite model is represented by a static set of weights and instructions, it can be directly stored in flash memory alongside the firmware components, resulting in an installable model size slightly smaller than 4 MB.

Table 2
Models characterization results.

Model	Measure	MotionSense	MobiAct	Size [MB]
Activity	Accuracy	0.993 ± 0.002	0.937 ± 0.001	10.7
Gender	Accuracy	0.997 ± 0.001	0.963 ± 0.010	205.4
Age [years]	MAE	0.386 ± 0.057	1.475 ± 0.132	8.6
Height [cm]	MAE	1.611 ± 0.196	4.091 ± 0.166	97.3
Weight [kg]	MAE	1.213 ± 0.143	4.747 ± 0.260	127.2

Table 3
Single attribute obfuscators performance on MotionSense dataset.

Obfuscator	Measure	Information	Before	After	Δr	RER
Gender	accuracy	Activity	0.993 ± 0.002	0.943 ± 0.003	−0.051	0.077
		gender	0.997 ± 0.001	0.542 ± 0.045	−0.456	0.915
Age [years]	Accuracy	Activity	0.993 ± 0.002	0.946 ± 0.003	−0.047	0.072
	MAE	Age	0.386 ± 0.057	116 ± 10	299.536	4.235
Height [cm]	Accuracy	Activity	0.993 ± 0.002	0.945 ± 0.002	−0.048	0.074
	MAE	Height	1.611 ± 0.196	129 ± 8	79.210	7.739
Weight [kg]	Accuracy	Activity	0.993 ± 0.002	0.946 ± 0.005	−0.048	0.073
	MAE	Weight	1.213 ± 0.143	90 ± 9	73.210	3.385

4.5. Measurement setup

To estimate the energy consumption of the IoT device, we measured the voltage drop across a sensing resistor (9.8Ω) placed in series with the power supply. The device was powered at 3.3 V through an NGMO2 Rohde & Schwarz dual-channel power supply [54]. We sampled the signals to be monitored during the experiments through a National Instruments NI-DAQmx PCI-6251 16-channel data acquisition board connected to a BNC-2120 shielded connector block [55,56].

5. Experiments and results

In the following, we present the experiments performed to characterize the effectiveness of preserving privacy in HAR and the energy efficiency of the proposed approach.

5.1. Privacy preservation performance

In the following, we present the experiments performed to characterize the effectiveness of preserving privacy in HAR and the energy efficiency of the proposed approach. First, we reported the recognition results obtained by the models, reported in Table 1, which represent the state-of-the-art HAR classification and regression. In particular, the proposed models are trained and tested using the original dataset divided into "train" and "test", where 75% are used as the training set and 25% as the test set. The training-testing procedure is performed five times with five different seeds of the random generator to have five different dataset splits to evaluate the robustness of the approach.

Table 2 reports the accuracy and the MAE calculated over MotionSense and MobiAct datasets together with the model size. The ability to recognize human activities and sensitive information appears very good for both datasets, with a reduction in performance in the case of MobiAct compared to MotionSense. In particular, the models can recognize human activity with an accuracy greater than 99% Moreover, it is undeniable that acquiring sensitive information with extraordinary accuracy is possible. In both datasets, we can estimate the gender of the subjects with over 96% accuracy. In contrast, for age, the average error barely reaches 1.5y in the worst case. Also, regarding height and weight, the average error is, respectively, around 4 cm and 5 kg, demonstrating how informative the data extracted from modern accelerometers and gyroscopes are and how important it is to be able to obscure them to preserve privacy. Notice that the reduced value of the standard deviation calculated over five different runs demonstrates the robust reproducibility of the results.

Starting from the reference models, we trained and tested four different autoencoders, each devoted to obfuscating a particular sensitive information. Tables 3 and 4 report the privacy preservation results of the single attribute obfuscators applied to the two available datasets. In summary, we can argue that the proposed approach obtains a very good performance of hiding sensitive information at the cost of a reduced accuracy drop in inferring human activities. Indeed, considering each single attribute obfuscator, the HAR accuracy decreases on average by about 4% and 2% for MotionSense and MobiAct, respectively. These results are almost identical to the decreases reported by Zhang et al. [21]. On the other hand, the obfuscation capability appears to be much more significant with a MAE of about 116y, 129 kg, and 90 cm, respectively, for age, weight, and height, corresponding to a Δr of about 299, 79, and 73 measured in MotionSense. Notice that Zhang et al. report about 9y, 59 kg, and 27 cm (Δr about 2, 7, and 4). MobiAct, even if with a slight reduction, it confirms these excellent results. Finally, analyzing the RER metrics, we can argue that the obfuscation introduced can be considered total given that the MAE always exceeds the average error committed by a random

Table 4
Single attribute obfuscators performance on MobiAct dataset.

Obfuscator	Measure	Information	Before	After	Δr	RER
Gender	Accuracy	Activity	0.937 ± 0.001	0.921 ± 0.007	−0.016	0.105
		gender	0.963 ± 0.010	0.623 ± 0.054	−0.353	0.754
Age [years]	Accuracy	Activity	0.937 ± 0.001	0.916 ± 0.006	−0.022	0.112
	MAE	Age	1.475 ± 0.132	96.734 ± 16.141	64.597	3.530
Height [cm]	Accuracy	Activity	0.937 ± 0.001	0.910 ± 0.006	−0.028	0.119
	MAE	Height	4.091 ± 0.166	109.321 ± 10.078	25.719	6.546
Weight [kg]	Accuracy	Activity	0.937 ± 0.001	0.917 ± 0.005	−0.018	0.107
	MAE	Weight	4.747 ± 0.260	79.449 ± 10.327	15.736	2.987

Table 5
Multiple attributes obfuscator performance on MotionSense dataset.

Information	Measure	Before	After	Δr	RER
Activity	Accuracy	0.993 ± 0.002	0.954 ± 0.002	−0.039	0.093
Gender	Accuracy	0.997 ± 0.001	0.515 ± 0.029	−0.483	0.969
Age [years]	MAE	0.386 ± 0.057	111 ± 13	288.562	4.080
Height [cm]	MAE	1.611 ± 0.196	99 ± 15	60.678	5.951
Weight [kg]	MAE	1.213 ± 0.143	68 ± 6	55.195	2.563

Table 6
Multiple attributes obfuscator performance on MobiAct dataset.

Information	Measure	Before	After	Δr	RER
Activity	Accuracy	0.937 ± 0.001	0.913 ± 0.008	−0.025	0.174
Gender	Accuracy	0.963 ± 0.100	0.600 ± 0.046	−0.378	0.801
Age [years]	MAE	1.475 ± 0.132	84.107 ± 15.976	56.035	3.070
Height [cm]	MAE	4.091 ± 0.166	79.069 ± 16.901	18.325	4.735
Weight [kg]	MAE	4.747 ± 0.260	70.023 ± 8.416	13.751	2.632

generator making the inference of individual sensitive information. Notice that, concerning the architecture of the autoencoders shown in Table 1, these results have been obtained using an intermediate autoencoder size that is 300-37-25-18-25-37-300 occupying 761 kB and 759 kB of memory in the case of dynamic and full integer 8-bit quantization, respectively.

Tables 5 and 6 report the obfuscation results obtained by a multiple attribute obfuscator on top of the two datasets. The results obtained in this case are excellent; however, we noticed a slight reduction for the single attribute obfuscators, demonstrating the robustness of the proposed approach. Concerning the autoencoder size, the reported results have been obtained using a larger configuration than that of the single attribute obfuscators. This result corresponds to the configuration 300-75-50-37-50-75-300, which needs about 958 kB for dynamic quantization and 955 kB of flash memory for full integer 8-bit quantization. Notice that the increase in size was necessary to keep the HAR performance unaltered for those of single attribute obfuscators.

To deeply analyze the obfuscation capabilities of the proposed method, the accelerometer and gyroscope signals of each dataset sample have been processed before and after the obfuscation to extract several synthetic features describing signal characteristics. Precisely, in this study, three sets of descriptors were calculated. The first set encompasses fundamental statistical descriptors that capture data trends and variations. These descriptors include: (i) mean, (ii) maximum value, (iii) standard deviation, and (iv) median. The second set encompasses *Hjorth* parameters, namely: (i) activity, (ii) mobility, and (iii) complexity. Lastly, the third set comprises *Kurtosis* and *Skewness* parameters, designed to encapsulate data characteristics related to shape. While the first set effectively describes sample tendencies, Hjorth parameters can capture the main characteristics of the signal in the frequency domain. Specifically, Hjorth activity represents the signal's power, the mobility of its mean frequency, and the complexity measures its change in frequency [57]. Kurtosis and Skewness are employed to depict the dispersion degree and symmetry of the data, respectively. In particular, Kurtosis measures whether the data exhibit heavy-tailed or light-tailed behavior compared to a normal distribution. In contrast, Skewness quantifies the extent of deviation from a perfectly symmetrical distribution [58].

The proposed 9 features have been calculated for each of the three channels of the two signals, resulting in a 54-dimensions feature space. The high-dimensional space has been embedded in a two-dimensional space using the t-distributed Stochastic Neighbor Embedding (t-SNE) techniques. The scatter plots of the sample labeled as "walking" and belonging from the MotionSense dataset are reported in Fig. 3 where female and male samples are drawn respectively in blue and red. The Figure highlights how, before the obfuscation, it was possible to appreciate the clear separation between the records belonging to the two genres while, after obfuscation, the same samples appear uniformly distributed over the plane. Moreover, before obfuscation, numerous clusters of samples appear evident, which presumably can be traced back to the different subjects. At the same time, these characteristics disappeared in the obfuscated data.

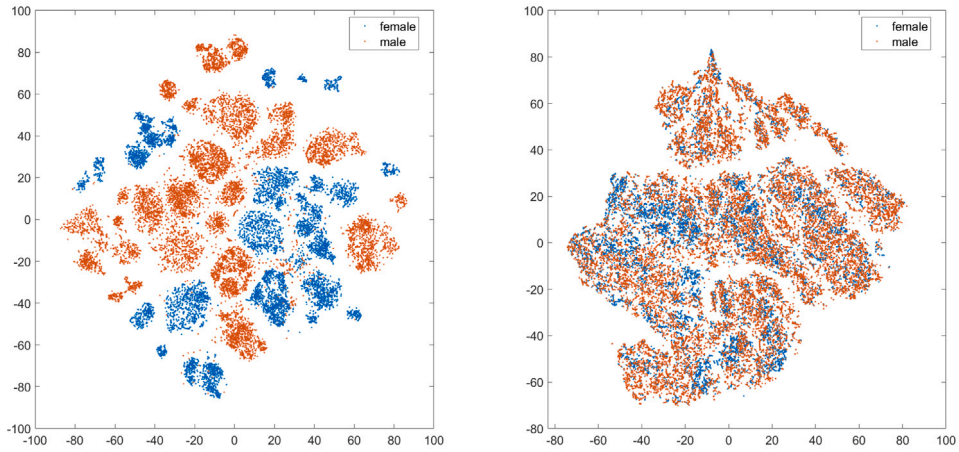


Fig. 3. Scatter plots of the "walking" samples characterized by means of 9 synthetic features before (a) and after (b) applying the obfuscator. The dimensionality reduction of the features has been obtained using the t-distributed Stochastic Neighbor Embedding technique.

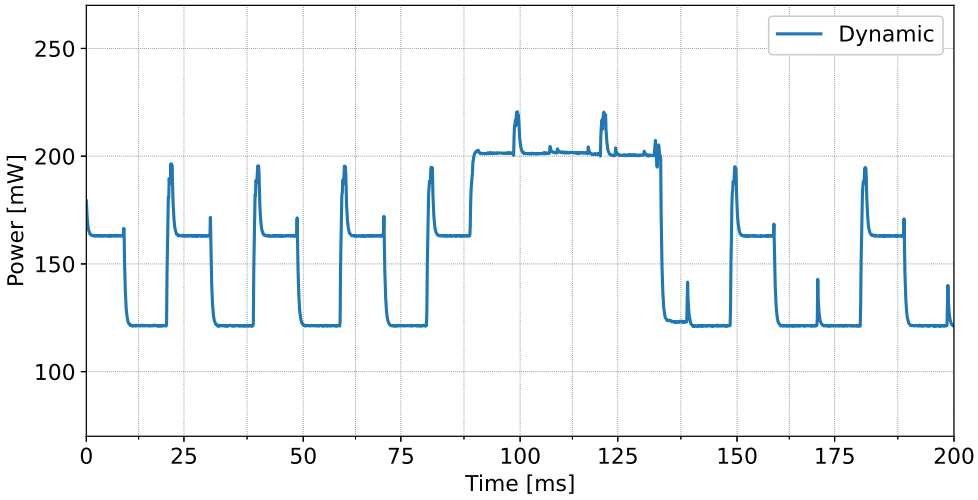


Fig. 4. Power consumption waveforms collected during the execution of the HAR application (subject to the autoencoder task) on the ESP32 device with power management enabled.

5.2. Energy efficiency

We conducted a deep characterization of energy and time inference of the proposed autoencoder on top of the ESP32 low-power IoT device, according to the measurement setup described in Section 4.5. Due to the memory constraints of the HW platform, we exploited the conversion on TensorFlow Lite representation and compression techniques such as dynamic and full integer 8-bit quantizations. For our experiments, we developed a sensor-based HAR application that continuously collects data from an MPU6050 integrated 6-axis motion tracking device that combines a 3-axis gyroscope and 3-axis accelerometer [53] and executes the autoencoder to remove sensitive information. It is worth mentioning that applying the simple lite conversion was insufficient to obtain a model small enough to be deployed on the ESP32 device, so compression techniques were required.

Considering the multiple attributes obfuscator, Fig. 4 shows the waveform of the power consumption of the device collected during the execution of the HAR application. The Figure highlights: (i) the power peaks related to sensor readings, (ii) the energy-saving mode during the sampling period between two consecutive samples, and (iii) the significant increase in power related to the execution of the obfuscator. It is worth mentioning that the waveform corresponds with the dynamic quantized autoencoder, and we did not report the waveform for the case of the full integer quantization because we did not find appreciable differences.

The energy and time needed to obfuscate data were characterized for the multiple attributes obfuscator when varying the autoencoder complexity as reported in Table 1. Fig. 5 shows the results in terms of energy consumption of each configuration of the

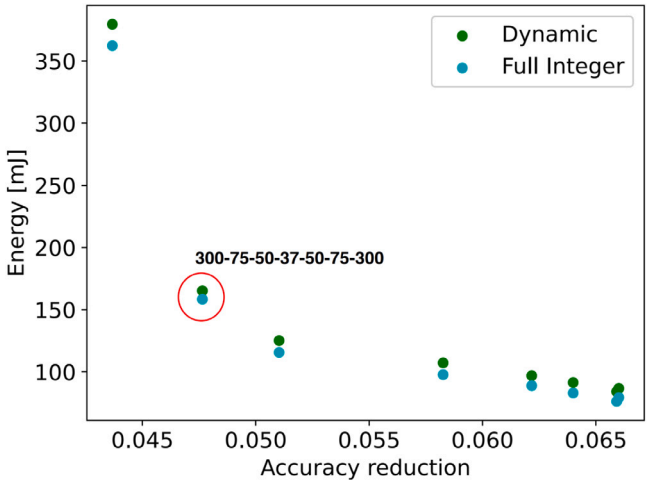


Fig. 5. Pareto curve reporting the accuracy reduction Vs the expected energy consumption.

Table 7
Characterization of the energy consumption and inference latency of the IoT device during the obfuscation phase (multi attributes).

Quantization	Obfuscation energy [mJ]	Obfuscation time [ms]
Dynamic	165.240	30.946 ± 0.005
Full Integer 8-bit	158.402	30.179 ± 0.027

multiple attributes obfuscator. In particular, the Figure shows the Pareto plane reporting the accuracy reduction plotted versus the energy consumption. With accuracy reduction, we mean the difference between the accuracy of the activity recognition without introducing any obfuscation mechanism and the accuracy after introducing the obfuscation mechanism. The two compression techniques provide comparable performance in terms of energy consumption, and the best trade-off is provided by the architecture size 300-75-50-37-50-75-300.

Table 7 reports, for the best configuration, the energy expenditure and the corresponding obfuscation time for both the case of dynamic and full integer 8-bit quantization, obtained by averaging 10 different execution bursts. As can be noticed, dynamic and full integer quantizations provide comparable performances: the energy consumption is low, around 165.240 mJ for the dynamic quantization and 158.402 mJ for the full integer 8-bit quantization. The obfuscation latency is in the order of 30 ms which is fully compatible with real-time requirements despite the low computing capability of the device.

6. Conclusion

Recognizing patterns of specific human activities has gained attention in many application domains, ranging from healthcare to smart homes, also because of an increasing diffusion of mobile and wearable devices equipped with many types of sensors (e.g., accelerometers, gyroscopes, magnetometers). Deep learning models designed to be executed on the available sensor signals allow inference regarding different activities at high accuracy levels. However, due to the energy and computation burden required by state-of-the-art neural network architectures, classification tasks are frequently delegated to edge or cloud computing devices, exposing users to leaks of sensitive information.

We presented in this work an approach for protecting user's privacy concerning the inference of sensitive attributes from sensor streams (i.e., weight, height, age, and gender) using a deep-learning autoencoder. This type of neural network is trained to transform the signal in a way that the information that could give rise to privacy breaches is filtered out. At the same time, the informative part concerning activity recognition is retained. The autoencoder architecture has been fine tuned to allow its porting on a low-power typical IoT platforms, and its energy consumption has also been measured to characterize the performance completely.

A comprehensive set of experimental results substantiates the capability of the proposed system of obfuscating the signal for the inference of sensitive attributes while keeping high accuracy levels in HAR tasks. Indeed, privacy-related inferences can be completely neutralized in the experimented set-up, with a modest (5% maximum) decrease in accuracy. At the same time, through its implementation on a low-power embedded device, we also showed the feasibility of the approach in typical wearable applications, with the best trade-off between loss of accuracy and energy consumption amounting to around a maximum of 165.240 mJ during the execution of the obfuscation task with an obfuscation latency of around 30 ms.

CRediT authorship contribution statement

Leonardo Bigelli: Validation, Software, Investigation, Data curation. **Chiara Contoli:** Writing – review & editing, Writing – original draft, Visualization, Investigation. **Valerio Freschi:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization. **Emanuele Lattanzi:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] K. Gopinath, L.P. Sai, A study on the positioning of the brand variants by smartwatch manufacturers: a technometrics approach, Technol. Anal. Strategic Manag. 35 (6) (2023) 689–703.
[2] Grand View Research, Wearable technology market size, share and trends analysis report by product, 2023, URL https://www.grandviewresearch.com/industry-analysis/wearable-technology-market. (Last Accessed 29 January 2024).
[3] T. Poongodi, R. Krishnamurthi, R. Indrakumari, P. Suresh, B. Balusamy, Wearable devices and IoT, in: A handbook of Internet of Things in biomedical and cyber physical system, Springer, 2020, pp. 245–273.
[4] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, Y. Liu, Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities, ACM Comput. Surv. 54 (4) (2021) 1–40.
[5] Z. Qian, Y. Lin, W. Jing, Z. Ma, H. Liu, R. Yin, Z. Li, Z. Bi, W. Zhang, Development of a real-time wearable fall detection system in the context of internet of things, IEEE Internet Things J. 9 (21) (2022) 21999–22007.
[6] Y. Liu, Y. Mu, K. Chen, Y. Li, J. Guo, Daily activity feature selection in smart homes based on pearson correlation coefficient, Neural Process. Lett. 51 (2020) 1771–1787.
[7] D. Talbot, The Era of Ubiquitous Listening Dawns, Tech. rep, MIT Technology Review, 2013, https://www.technologyreview.com/2013/08/08/177068/the-era-of-ubiquitous-listening-dawns/.
[8] R. Aloufi, H. Haddadi, D. Boyle, Paralinguistic privacy protection at the edge, ACM Trans. Privacy Secur. 26 (2) (2023) 1–27.
[9] Y. Iwasawa, K. Nakayama, I. Yairi, Y. Matsuo, Privacy issues regarding the application of DNNs to activity-recognition using wearables and its countermeasures by use of adversarial training, in: IJCAI, 2017, pp. 1930–1936.
[10] A. Jain, V. Kanhangad, 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, 2016, pp. 597–602.
[11] L. Zhang, W. Cui, B. Li, Z. Chen, M. Wu, T.S. Gee, Privacy-preserving cross-environment human activity recognition, IEEE Trans. Cybern. (2021).
[12] E. Antwi-Boasiako, S. Zhou, Y. Liao, Q. Liu, Y. Wang, K. Owusu-Agyemang, Privacy preservation in distributed deep learning: A survey on distributed deep learning, privacy preservation techniques used and interesting research directions, J. Inf. Secur. Appl. 61 (2021) 102949.
[13] J. Liu, Z. Zhao, P. Li, G. Min, H. Li, Enhanced embedded AutoEncoders: An attribute-preserving face de-identification framework, IEEE Internet Things J. (2023).
[14] D. Bank, N. Koenigstein, R. Giryes, Autoencoders, 2020, arXiv preprint arXiv:2003.05991.
[15] G. Schram, R. Wang, K. Liang, Using autoencoders on differentially private federated learning GANs, 2022, arXiv preprint arXiv:2206.12270.
[16] M. Malekzadeh, R.G. Clegg, A. Cavallaro, H. Haddadi, Mobile sensor data anonymization, in: Proceedings of the International Conference on Internet of Things Design and Implementation, 2019, pp. 49–58.
[17] M. Malekzadeh, R.G. Clegg, A. Cavallaro, H. Haddadi, Privacy and utility preserving sensor-data transformations, Pervasive Mob. Comput. 63 (2020) 101132.
[18] N. Raval, A. Machanavajjhala, J. Pan, Olympus: Sensor privacy through utility aware obfuscation, Proc. Priv. Enhancing Technol. 2019 (1) (2019) 5–25.
[19] A. Boutet, C. Frindel, S. Gambs, T. Jourdan, R.C. Ngueveu, DYSAN: Dynamically sanitizing motion sensor data against sensitive inferences through adversarial networks, in: Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security, 2021, pp. 672–686.
[20] L. Lyu, X. He, Y.W. Law, M. Palaniswami, Privacy-preserving collaborative deep learning with application to human activity recognition, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 1219–1228.
[21] D. Zhang, L. Yao, K. Chen, Z. Yang, X. Gao, Y. Liu, Preventing sensitive information leakage from mobile sensor signals via integrative transformation, IEEE Trans. Mob. Comput. 21 (12) (2021) 4517–4528.
[22] A. Garain, R. Dawn, S. Singh, C. Chowdhury, Differentially private human activity recognition for smartphone users, Multimedia Tools Appl. 81 (28) (2022) 40827–40848.
[23] P. Climent-Pérez, F. Florez-Revuelta, Privacy-preserving human action recognition with a many-objective evolutionary algorithm, Sensors 22 (3) (2022) 764.
[24] S. Menasria, M. Lu, A. Dahou, PGAN framework for synthesizing sensor data privately, J. Inf. Secur. Appl. 67 (2022) 103204.
[25] M. Malekzadeh, R.G. Clegg, H. Haddadi, Replacement autoencoder: A privacy-preserving algorithm for sensory data analysis, 2017, arXiv preprint arXiv:1710.06564.
[26] M. Malekzadeh, R.G. Clegg, A. Cavallaro, H. Haddadi, Protecting sensory data against sensitive inferences, in: Proceedings of the 1st Workshop on Privacy By Design in Distributed Systems, 2018, pp. 1–6.
[27] P. Delgado-Santos, R. Tolosana, R. Guest, R. Vera-Rodriguez, F. Deravi, A. Morales, GaitPrivacyON: Privacy-preserving mobile gait biometrics using unsupervised learning, Pattern Recognit. Lett. 161 (2022) 30–37.
[28] S. Abbasi, M. Famouri, M.J. Shafiee, A. Wong, OutlierNets: highly compact deep autoencoder network architectures for on-device acoustic anomaly detection, Sensors 21 (14) (2021) 4805.
[29] H. Ren, D. Anicic, T.A. Runkler, Tinyol: Tinyml with online-learning on microcontrollers, in: 2021 International Joint Conference on Neural Networks, IJCNN, IEEE, 2021, pp. 1–8.

[30] A. Alsalemi, Y. Himeur, F. Bensaali, A. Amira, An innovative edge-based internet of energy solution for promoting energy saving in buildings, Sustainable Cities Soc. 78 (2022) 103571.
[31] D.-V. Bratu, R.Ş.T. Ilinoiu, A. Cristea, M.-A. Zolya, S.-A. Moraru, Anomaly detection using edge computing AI on low powered devices, in: IFIP International Conference on Artificial Intelligence Applications and Innovations, Springer, 2022, pp. 96–107.
[32] A. Sathiamoorthy, S. Mithusan, R. Rathnayaka, S. Kajenthiran, M.M. Hansika, D. Pandithage, StreamSafe: Improving QoS and security in IoT networks, Int. Res. J. Innovat. Eng. Technol. 7 (11) (2023) 170.
[33] S.S. Hammad, D. Iskandaryan, S. Trilles, An unsupervised tinyml approach applied to the detection of urban noise anomalies under the smart cities environment, Internet Things 23 (2023) 100848.
[34] H.V. Düdükçü, M. Taşkiran, ECG data anomalies detection with stacked autoencoder on low power and low memory microcontrollers, Curr. Res. Eng. (2023).
[35] E. Oliver, R. Yue, A. Dutta, A secure vitals monitoring point-of-care device, in: 2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society, EMBC, IEEE, 2023, pp. 1–4.
[36] W. Lin, M. Xu, J. He, W. Zhang, Privacy, security and resilience in mobile healthcare applications, Enterpr. Inf. Syst. 17 (3) (2023) 1939896.
[37] Q. Lin, Developing Wearable Applications with Innovative Sensing Modalities for Human Activity Recognition and Key Generation (Ph.D. thesis), UNSW Sydney, 2020.
[38] M.A. Jan, F. Khan, R. Khan, S. Mastorakis, V.G. Menon, M. Alazab, P. Watters, Lightweight mutual authentication and privacy-preservation scheme for intelligent wearable devices in industrial-CPS, IEEE Trans. Ind. Inf. 17 (8) (2020) 5829–5839.
[39] E. Lattanzi, M. Donati, V. Freschi, Exploring artificial neural networks efficiency in tiny wearable devices for human activity recognition, Sensors 22 (7) (2022) 2637.
[40] C. Contoli, E. Lattanzi, A study on the application of TensorFlow compression techniques to human activity recognition, IEEE Access 11 (2023) 48046–48058, http://dx.doi.org/10.1109/ACCESS.2023.3276438.
[41] Tensorflow official website, 2023, http://www.tensorflow.org/?hl=en. (Accessed: 10 July 2023).
[42] TensorFlow Lite for microcontrollers, 2023, http://www.tensorflow.org/lite/microcontrollers?hl=en. (Accessed: 10 July 2023).
[43] Espressif, ESP32-C3-WROOM-02 datasheet, 2022, URL https://www.espressif.com/en/support/documents/technical-documents. (last accessed 22 June 2023).
[44] N. Ketkar, Introduction to keras, in: Deep Learning with Python, Springer, 2017, pp. 97–111.
[45] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, A. Talwalkar, Hyperband: A novel bandit-based approach to hyperparameter optimization, J. Mach. Learn. Res. 18 (1) (2017) 6765–6816.
[46] M. Malekzadeh, R.G. Clegg, A. Cavallaro, H. Haddadi, Mobile sensor data anonymization, in: Proceedings of the International Conference on Internet of Things Design and Implementation, IoTDI '19, ACM, New York, NY, USA, 2019, pp. 49–58, http://dx.doi.org/10.1145/3302505.3310068, URL http://doi.acm.org/10.1145/3302505.3310068.
[47] G. Vavoulas, C. Chatzaki, T. Malliotakis, M. Pediaditis, M. Tsiknakis, The mobiact dataset: Recognition of activities of daily living using smartphones, in: International Conference on Information and Communication Technologies for Ageing Well and E-Health, Vol. 2, SciTePress, 2016, pp. 143–151.
[48] O. Banos, J.-M. Galvez, M. Damas, H. Pomares, I. Rojas, Window size impact in human activity recognition, Sensors 14 (4) (2014) 6474–6499.
[49] J. Cheng, O. Amft, P. Lukowicz, Active capacitive sensing: Exploring a new wearable sensing modality for activity recognition, Int. Conf. Pervasive Comput. (2010) 319–336.
[50] M.M. Hassan, M.Z. Uddin, A. Mohamed, A. Almogren, A robust human activity recognition system using smartphone sensors and deep learning, Future Gener. Comput. Syst. 81 (2018) 307–313.
[51] C. Hou, A study on IMU-based human activity recognition using deep learning and traditional machine learning, in: 2020 5th International Conference on Computer and Communication Systems, ICCCS, IEEE, 2020, pp. 225–234.
[52] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manag. 45 (4) (2009) 427–437.
[53] InvenSense Inc., MPU-6050 product specification, 2023, URL https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6050/. (Last Accessed 22 June 2023).
[54] Rohde&Schwarz, NGMO2 datasheet, 2020, URL https://www.rohde-schwarz.com/it/brochure-scheda-tecnica/ngmo2/. (Last Accessed 22 June 2023).
[55] National.Instruments, PC-6251 datasheet, 2020, URL http://www.ni.com/pdf/manuals/375213c.pdf. (Last Accessed 22 June 2023).
[56] National.Instruments, Installation guide BNC-2120, 2020, URL http://www.ni.com/pdf/manuals/372123d.pdf. (Last Accessed 22 June 2023).
[57] B. Hjorth, EEG analysis based on time domain properties, Electroencephalogr. Clin. Neurophysiol. 29 (3) (1970) 306–310.
[58] T.-H. Kim, H. White, On more robust estimation of skewness and kurtosis, Finance Res. Lett. 1 (1) (2004) 56–73.