Preventing Sensitive Information Leakage from Mobile Sensor Signals via Integrative Transformation

Dalin Zhang Member, IEEE, Lina Yao Member, IEEE, Kaixuan Chen , Zheng Yang Senior Member, IEEE, Xin Gao Member, IEEE and Yunhao Liu Fellow, IEEE

Abstract—Ubiquitous mobile sensors on human activity recognition pose the threat of leaking personal information that is explicitly contained within the time-series sensor signals and can be extracted by attackers. Existing protective methods only support specific sensitive attributes and require massive relevant sensitive ground truth for training, which is unfavourable to users. To fill this gap, we propose a novel data transformation framework for prohibiting the leakage of sensitive information from sensor data. The proposed framework transforms raw sensor data into a new format, where the sensitive information is messed up and the task information (e.g., human activities) is retained. Training can be conducted without using any personal information as ground truth. Meanwhile, all attributes of sensitive information (e.g., age, gender) can be hidden through a one-time transformation collectively. The experiment results on two multimodal sensor-based human activity datasets manifest the feasibility of the presented framework in concealing users’ sensitive information without degrading the usability of the data for activity recognition.

Index Terms—mobile sensors, human activity recognition, sensitive information protection, neural network

1 INTRODUCTION

HUMAN activity recognition (HAR) plays an important role in various attractive human-in-the-loop applications especially in the smart living scenario [1], [2]. A typical case is the smart health-care, where wearable sensors are employed to capture the motions of users and a health-care provider can use the data to support exercise and medical suggestions or emergency rescues. Our lives become safer and more convenient with the assist of these personalized and ubiquitous services. Nevertheless, the concern of privacy leakage of this kind of personal data has drawn an increasing attention in recent years [3]. Although the sensors are originally used to capture the movement of users, personal traits can also be hold within the continuous signals unintentionally.

Consider a scenario where an elderly subject lives alone that constant monitoring his/her health situation is required. Smart wearable devices or smartphones with multiple built-in sensors (e.g., accelerometer, gyroscope, and magnetometer) are usually used for the monitoring purpose. The collected data is continuously transmitted to the health-care institute for analysis. An essential task of the analysis is to recognize the activities of the user, such as walking, jogging, and running. However, people perform activities in different ways because of the divergence of personal facts like age, gender, and weight. For example, a subject with high body weight would walking slowly than a subject with lower body weight. Therefore, the subject’s weight information could be determined through interpreting the sensor signals. Other personal information like age, gender, and height can also be figured out in the similar way [3]–[5]. This kind of information leakage is unacceptable so that the sensor data cannot be directly sent out without any privacy destruction procedure. On the contrary, the inference about human activities such as walking, jogging, and running is the purpose of collecting the sensor data and extremely critical for the following applications especially when for the healthcare treatment. As a result, the sensor data should not be modified to avoid degrading the activity recognition performance. In this context, we draw a conundrum, where the sensor data should not be released unchangeably to avoid privacy leakage whereas modification should also be avoided for keep the recognition precision.

Considering this contradiction, we report a data transformation approach that manages to separate the two sorts of information contained within the same sensor data and then blur the user sensitive information and maintain the activity information at the same time. Figure 1 depicts the general data transformation framework for hindering leaking sensitive information. Ideally, sensitive information is hidden while the desired task information can pass through during the transformation process. However, as all information (i.e., sensitive and task) is enclosed in the same raw data, it is impractical to remove sensitive information yet to keep all the other information unchanged at the same time. In light of this fact, we make an assumption that the specific task, human activity recognition, is the only desired task that needs to be kept unaffected in this study. Nevertheless, our reported framework is applicable to preventing sensitive information leakage with any other certain desired tasks or a multi-task scenario, where more
than one desired information needs to be untouched.

There are some factors that make the preventing sensitive information leakage task challenging. First, as mentioned above, the sensitive information and desired information are both from the same chunk of data that it is painful to modify one in isolation from the other one [6]. Moreover, the desired information is the basis to the following applications, thus the impact on it should be minimized. Second, the inherent noises encompassed in the sensor signals deteriorate the data quality severely [7]. In addition, different sensing modes embed information in different ways that developing a unified sensitive information protection approach for all modes is full of challenges, especially for traditional feature engineering. Existing solutions can be divided into two main streams: (1) ceasing sensors based on pre-defined conditions or user specifications, and (2) modifying raw data to remove sensitive information before being sent out to a service provider. The first solution is non-intelligent that are only damaging to the following applications yet not able to fully prevent information leakage at all time [8], [9]. In contrast, the latter solution is a more practical and promising approach that engages boosting attentions [3], [10], [11].

As illustrated in Figure 1, the main idea of the data transformation solution is to seek an data conversion algorithm that makes a better trade-off between smearing the sensitive information and retaining the desired information. Current works have two principle drawbacks in this context. One the one hand, most approaches can merely protect one specific sensitive information, such as gender [11] or user ID [3]. This kind of methods would fail when there are multiple sensitive information need to be protected in most practical scenarios. On the other hand, during the preparation of the data transformation model, users’ sensitive information that needs to be protected is required for training [3], [10], [11]. This requirement induces a new source of sensitive information leakage and is more severe than revealing the sensor data since the users’ personal information is disclosed directly.

Targeting the above defects, this paper presents an integrative data transformation framework that can protect all sensitive information at a one-time transformation and does not require users to provide any personal attributes during framework preparation. Inspired by the image style transformation research [12], we separate the sensitive information and desired information logically into “content” information and “style” information, although they are physically contained within the same block of data. For one thing, a user’s personal attributes, like gender or weight, influence how he/she performs an activity task so we this information as “style” information. As an illustration, a subject with a higher body weight commonly moves slowly than with a low body weight. Besides, the desired information typically requires what a user is doing so can be accounted for “content” information. In this context, our framework aims to transform the data “style” comparable to random noise so that the sensitive information is blurred, while to keep the data “content” exact as raw data. Concretely, we design a fully convolutional TransNet that is responsible to carry out the data transformation process and an auxiliary LossNet for defining the training targets of the TransNet. The LossNet determines a style loss and a content loss that try to minimize the differences between the transformed data and random noises as well as raw data, respectively. During the training process, none of user sensitive information is required and only sensor data is presented. A preliminary version of this work has been reported in [13]. Experiments are conducted on two multimodal activity recognition datasets to protect five types of sensitive information (i.e., age, gender, height, weight, and ID). The empirical validation results manifest that the reported approach can successfully hide all sensitive information simultaneously at one-time transformation while supporting a high preservation level of desired information with regard to activity recognition accuracy.

2 RELATED WORK

2.1 Activity Recognition from Mobile Sensor Signals

The mobile sensor-based HAR aims to recognize human activities from multimodal time-series signals that originated from mobile sensors. In this context, machine learning technologies have shown dominant performance since it is hard to uncover the latent traits of sensor signals and their complex correlations by computing methods. Popular methods like Support Vector Machine (SVM) [14] have demonstrated effective in dealing with subject-dependent scenarios. However, they get into trouble when facing the subject-independent applications [15]. Recent years, deep neural networks have been proven superior to traditional machine learning approaches in various fields including the HAR. They are first applied to mobile sensor-based human activity recognition by [16]. Restricted Boltzmann Machines (RBM) is leveraged for automatic feature extraction and demonstrated significant performance improvement over classical feature engineering. In addition to unsupervised feature learning with deep neural networks, supervised learning with deep neural networks were much more widely explored by researchers. Such as in [17], [18], convolutional neural networks were adopted to fuse different sensing modalities and thus enhanced activity recognition. The advanced attention-based mechanisms were introduced by [19], [20] to achieve remarkable recognition accuracy on different evaluation datasets. The attention-based methods could also provide explicable features of neural networks for HAR [19]. Chen et al. proposed a multi-agent attention model to extract attentive features from different angles and achieved promising performance on several public datasets [21].

2.2 Prevent Sensitive Information Leakage from Mobile Sensor Signals

The main application of human activity recognition is to monitor human behaviors in a smart environment [22], [23]. Thus the wearable sensors need to capture the physiological signals of users continuously. Since the way an activity is taken varies among users (due to age, gender, and weight) [24], an adversary could infer
user this sensitive information through the time series signals [3]. Although deep neural networks demonstrate powerful abilities in mobile sensor data analysis, the privacy concern limits their further applications.

One naive way of protecting user sensitive information is to stop sensors working based on predefined settings or user specifications [8], [9], [25], [26], for example, Olejnik et al. [26] proposed an automatic runtime control based on a smartphone usage context. However, this approach dramatically influences the application usage, especially for health monitoring and elderly care. Adding random noise to sensor data is an alternative way to hiding sensitive information [27]–[29]. This kind of methods works well for preventing sensitive information. However, it destroys the usability of data severely because the noise usually applies undifferentiated perturbations to all information, including both target and sensitive ones [27]. A more advanced approach is to transform raw sensor signals into a new representation without user sensitive information embedded, but with data usability kept. For example, the authors of [3] proposed an adversarial training strategy that transformed raw sensor data into neural network features to eliminate user-discriminative information. The authors of [11] reported transforming raw data into a new representation with the same data size to eliminate the gender information. Our goal is to find an optimal transformation function that can be drawn normally:

\[ f^*(S_d) = Y_s \]

As shown in Figure 2, in order to achieve such a transformation function \( f(.) \), we propose a framework comprising a TransNet \( f(.) \) that performs the data transformation process, and a LossNet \( \phi \) that defines several loss functions for training the TransNet to get rid of the sensitive information from raw sensor data. Specifically, the LossNet defines: a “style” loss that measures the “style” difference between the transformed data and random noise, a “content” loss that measures the “content” difference between transformed data and raw data, and a usability loss that specifically helps to keep the inference accuracy of the target information.

Each loss function computes a scalar value \( \ell_i(\hat{S}_d, Y_i) \) measuring the difference between the transformed data \( \hat{S}_d \) and a target \( Y_i \) (e.g. random noise or raw data). The TransNet is trained with the stochastic gradient descent to minimize the weighted combination of all loss functions:

\[
\arg\min E_i \sum_{i=1}^{n} \lambda_i \ell_i(f(S_d), Y_i), \quad \sum_{i=1}^{n} \lambda_i = 1. \quad (1)
\]

3 Methodology

3.1 Problem Statement and Definition

As multiple sensing modalities are usually used in HAR, we first assume all sensor signals are synchronized and recorded at a same frequency. Let \( X(t) = [x_1(t), x_2(t), ..., x_m(t)] \) be the reading values of \( m \) sensor components (each component could be an axis of a mobile sensor) recorded at the time step \( t \). Thus for a time period of \( d \) in length starting from time \( t \), we have time series sensor signals \( S_d(t) = [X(t); X(t+1); ...; X(t+d-1)] \). For simplicity, we use \( S_d \) instead of \( S_d(t) \) in the rest of the paper.

The \( S_d \) is two-dimensional (2D) raw sensor data with one dimension representing time and the other representing sensor components. In traditional conditions, a service provider uses a human activity inference function \( I_a(.) \) to recognize users’ activities \( Y_a \) from raw sensor data \( S_d \) for subsequent analysis. Ideally, \( I_a(S_d) = Y_a \). In the meantime, there exists a certain sensitive inference function \( I_s(.) \) that can be used to infer user sensitive information \( Y_s \) (such as gender and age) from the raw sensor data \( S_d \) as well. Ideally, \( I_s(S_d) = Y_s \). Our goal is to find an optimal transformation function \( f^*(.) \), so that the sensitive inference cannot be drawn from the optimal transformed data \( \hat{S}_d = f^*(S_d) \): \( I_s(\hat{S}_d) \neq I_s(S_d) = Y_s \), whereas the target inference about human activities \( I_a(\hat{S}_d) \) can be drawn normally: \( I_a(\hat{S}_d) = I_a(S_d) = Y_a \). Here \( \hat{S}_d \) is the transformed data which is transformed from raw data \( S_d \) by a transformation function \( f(.) \) and \( S_d^a \) is the optimal transformed results from where the sensitive information cannot be inferred.

3.2 Overview

The goal of this study is to transform raw data before being revealed to an untrusted service provider so that the sensitive information cannot be inferred from the transformed data. In contrast, the target information can be inferred as usual. To this end, we propose to transform raw data into a new representation that has a “style” (sensitive information) of random noise and a “content” (target formation) of raw data. The transformed data should satisfy such conditions: when an adversary tries to infer user sensitive information from the transformed data, the results should be as unreliable as drawn from random noise; whereas a service provider can make inferences about target information from the transformed data with as high accuracy as from the raw data.

As shown in Figure 2, in order to achieve such a transformation function \( f(.) \), we propose a framework comprising a TransNet \( f(.) \) that performs the data transformation process, and a LossNet \( \phi \) that defines several loss functions for training the TransNet to get rid of the sensitive information from raw sensor data. Specifically, the LossNet defines: a “style” loss that measures the “style” difference between the transformed data and random noise, a “content” loss that measures the “content” difference between transformed data and raw data, and a usability loss that specifically helps to keep the inference accuracy of the target information.

Each loss function computes a scalar value \( \ell_i(\hat{S}_d, Y_i) \) measuring the difference between the transformed data \( \hat{S}_d \) and a target \( Y_i \) (e.g. random noise or raw data). The TransNet is trained with the stochastic gradient descent to minimize the weighted combination of all loss functions:

\[
\arg\min E_i \sum_{i=1}^{n} \lambda_i \ell_i(f(S_d), Y_i), \quad \sum_{i=1}^{n} \lambda_i = 1. \quad (1)
\]

3.3 Network Structure

3.3.1 LossNet

The LossNet \( \phi \) is a traditional 2D convolutional neural network for human activity recognition. It is first trained from scratch on raw training sensor data and then fixed for the subsequent training process of the TransNet. The detailed configuration of the LossNet is depicted in Table 1. The LossNet has two convolutional blocks, each of which has two convolutional layers and a maxpooling layer. The input into the LossNet has a size of \( m \times d \times 1 \), where \( m \) is the number of sensor components, \( d \) is time period length, and \( 1 \) is the number of feature maps. The period length of \( d \) is set to 50 in this study. The convolutional kernel is always set to \( 1 \times 3 \), and the maxpooling is always applied along the time dimension to reduce the feature map size by half. After flattening, the output of the second pooling layer is fed into a dense layer of size 400. As last, a dense layer with the softmax activation function defined as \( \text{softmax}(x_i) = \frac{1}{Z} \exp(x_i) \) with \( Z = \sum \exp(x_i) \), is appended for the final output. The loss function for training the LossNet is a cross-entropy loss for human activity classification:

\[
\ell^\phi_a = - \sum_e Y_{a,e} \log(\phi(S_d) e), \quad (2)
\]

where \( Y_{a,e} \) and \( \phi(S_d) e \) is the label and the predicted probability of the activity category \( e \), respectively. The predicted probability \( \phi(S_d) e \) is output from the LossNet with the raw data as input.

3.3.2 TransNet

The TransNet \( f(.) \) is a fully convolutional neural network with downsampling first and upsampling to the original size afterward. The fully convolutional TransNet can take any size of data as
interpolating upsampling, deconvolution allows the upsampling to upsample to the original size. Rather than depending on an inferences they make are not reliable, and (2) we would like data is raw or transformed when they only access the revealed data, lies in two folds: (1) the adversary cannot tell whether the revealed data and raw data. We also define a usability loss to keep the inference accuracy of the target information explicitly. The TransNet is trained through minimizing a weighted combination of the above loss functions to prevent sensitive information leakage while simultaneously preserving the target information.

The configurations of the LossNet and TransNet. $H$, $W$, $F$, $sH$, and $sW$ refer to height, width, the number of feature maps, stride height, and stride width respectively.

### Table 1

<table>
<thead>
<tr>
<th>Layer</th>
<th>Input Size (H x W x F)</th>
<th>Kernel/Stride (H x W x H x W)</th>
<th>Padding</th>
<th>Activation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conv1_1</td>
<td>m x 50 x 1</td>
<td>1 x 3/1 x 1</td>
<td>Same</td>
<td>Relu</td>
</tr>
<tr>
<td>Conv1_2</td>
<td>m x 50 x 16</td>
<td>1 x 3/1 x 1</td>
<td>Same</td>
<td>Relu</td>
</tr>
<tr>
<td>MaxPool1</td>
<td>m x 50 x 16</td>
<td>1 x 2/1 x 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conv2_1</td>
<td>m x 25 x 32</td>
<td>1 x 3/1 x 1</td>
<td>Same</td>
<td>Relu</td>
</tr>
<tr>
<td>Conv2_2</td>
<td>m x 25 x 32</td>
<td>1 x 3/1 x 1</td>
<td>Same</td>
<td>Relu</td>
</tr>
<tr>
<td>MaxPool2</td>
<td>m x 25 x 32</td>
<td>1 x 2/1 x 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dense</td>
<td>flat(m x 12 x 32)</td>
<td></td>
<td></td>
<td>Relu</td>
</tr>
<tr>
<td>Dense</td>
<td>400</td>
<td></td>
<td></td>
<td>softmax</td>
</tr>
<tr>
<td>Conv1</td>
<td>m x 50 x 1</td>
<td>1 x 3/1 x 1</td>
<td>Same</td>
<td>Relu</td>
</tr>
<tr>
<td>MaxPool1</td>
<td>m x 50 x 16</td>
<td>1 x 2/1 x 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conv2</td>
<td>m x 25 x 16</td>
<td>1 x 3/1 x 1</td>
<td>Same</td>
<td>Relu</td>
</tr>
<tr>
<td>MaxPool1</td>
<td>m x 25 x 32</td>
<td>1 x 2/1 x 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DeConv1</td>
<td>m x 13 x 32</td>
<td>1 x 3/1 x 1</td>
<td>Same</td>
<td>Relu</td>
</tr>
<tr>
<td>Conv4</td>
<td>m x 26 x 32</td>
<td>1 x 3/1 x 1</td>
<td>Same</td>
<td>Relu</td>
</tr>
<tr>
<td>DeConv1</td>
<td>m x 26 x 32</td>
<td>1 x 3/1 x 2</td>
<td>Same</td>
<td>Relu</td>
</tr>
<tr>
<td>Conv4</td>
<td>m x 52 x 32</td>
<td>1 x 3/1 x 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

process to be learned jointly with the rest of the network. Although the input and output have the same size, there are several benefits to the networks that first downsample and then upsample. The first benefit is the computational cost, which is positively correlated with the size of feature maps [30]. The second benefit comes from the receptive field sizes. With downsampling by a factor of $D$, each additional convolutional layer increases the receptive field size by $2D$ without extra computational cost. Otherwise, it only increases the receptive field size by 2 [31]. The input and output of the TransNet both have a size of $m \times 50 \times 1$. To achieve the size unchanged, we tune the padding options of both the first pooling layer and the last convolutional layer to “Valid”. The kernel size setting is consistent with the LossNet. The final convolutional layer of TransNet has no activation functions to make the transformed data fake the raw data, which has both positive and negative values.

### 3.4 “Style” and “Content” Consistency

#### 3.4.1 Content Consistency

We define a “content” loss function for measuring the “content” consistency between the transformed data and raw data. The “content” information describes what a user does during the data recording period $d$, which is human activities in this study. As deeper layers help extract better features, we encourage the raw data $S_d$ and the transformed data $\hat{S}_d$ to have similar feature representations as computed by a deeper convolutional layer of the LossNet $\phi$. Formally, let $\phi_j(S_d)$ and $\hat{\phi}_j(S_d)$ be the outputs of the $j$th layer of the network $\phi$ when the input of $\phi$ is the transformed data $\hat{S}_d$ and raw data $S_d$ respectively. If $j$ is a convolutional layer, then $\phi_j(S_d)$ will be feature maps of shape $C_j \times H_j \times W_j$. The “content” difference of layer $j$ is defined as the Euclidean distance between the feature representations of the transformed data $\hat{S}_d$ and raw data $S_d$:

$$
\ell_{c,j}^{\phi} = \frac{1}{C_j H_j W_j} \| \phi_j(S_d) - \hat{\phi}_j(S_d) \|^2.
$$

input, which is another advantage of our framework. The reason we resize the transformed data into the same size as the raw data lies in two folds: (1) the adversary cannot tell whether the revealed data is raw or transformed when they only access the revealed data, so any inferences they make are not reliable, and (2) we would like to keep as abundant information as possible.

The detailed configuration of the TransNet is also illustrated in Table 1. We use two convolution/maxpooling pairs to downsample the input data followed by two deconvolution/convolution blocks to upsample to the original size. Rather than depending on an interpolating upsampling, deconvolution allows the upsampling...
We use the “content” difference of the layer $Conv_{2\_2}$ of the LossNet to produce the “content” loss:

$$
\ell^c_u = -\sum c Y_{a,c} \log (\hat{\phi}(\hat{S}_d)_c),
$$

(8)

where $Y_{a,c}$ and $\phi(\hat{S}_d)_k$ are the label and the predicted probability of the activity category $k$, respectively. The activity predicted probability $\phi(\hat{S}_d)_k$ is output from the pretrained LossNet with the transformed data as input.

The final loss function is the weighted summation of all individual losses $\ell^c_c$, $\ell^\phi_u$, and $\ell^\phi_s$:

$$
\ell^o = \mathbb{E}[\lambda^c_u \ell^c_u(\hat{S}_d, S_d) + \lambda^\phi_u \ell^\phi_u(\hat{S}_d, N_s) + \lambda^\phi_s \ell^\phi_s(\hat{S}_d, Y_o)] .
$$

(9)

The weight of each loss $\lambda_i$ is set experimentally and kept adding up to 1.

3.5 Training Process

The LossNet $\phi$ is first trained from scratch on raw training data for inferring target information that is human activity in this study and then fixed during the subsequent training process of the TransNet. Note that when training the LossNet, only raw training data and the corresponding labels of human activities are provided; the labels of user sensitive information are not required. After training the LossNet, we start to train the TransNet. The goal of training the TransNet is to let the transformed data have a “style” of random noise and a “content” of raw data. Thus, the transformed data $\hat{S}_d$, raw data $S_d$ and random noise $N_s$ are input into the pretrained LossNet, respectively. Then the final loss calculated by Eq. (9) is obtained, which at last the TransNet is trained to minimize.

Algorithm 1 describes the detailed training process of our proposed data transformation mechanism. It only takes raw training data $S_d$, and its paired human activity ground truth $Y_o$ as input. The training process comprises training both the LossNet and the TransNet separately. In contrast, only the TransNet is used to take raw test data as input, and outputs transformed test data during the test stage.

Algorithm 1 Training Process of the Data Transformation Framework for Multiple Sensitive Information Protection

**Input:** raw training data $S_d$, human activity labels $Y_o$

**Pretrain LossNet $\phi$**

1: LossNet $\phi \leftarrow$ random initializing
2: Train LossNet $\phi \leftarrow \ell^o_u = -\sum_c Y_{a,c} \log (\phi(S_d)_c)
3: Fix weights and biases of LossNet $\phi$

**Train TransNet $f(.)$**

1: TransNet $f(.) \leftarrow$ random initializing
2: Generate transformed training data $\hat{S}_d = f(S_d)$
3: Generate random noise $N_s$
4: Feed transformed training data $\hat{S}_d$, random noise $N_s$, raw data $S_d$ into pretrained LossNet $\phi$ respectively
5: Calculate “content” loss $\ell^c_o = ||\phi(\hat{S}_d) - \phi(S_d)||_2^2$, “style” loss $\ell^\phi_o = \sum_j ||G_j(\hat{S}_d) - G_j(N_s)||_2^2$, and usability loss $\ell^\phi_u = - \sum_k Y_{a,k} \log (\phi(\hat{S}_d)_k)$, where $G(.)$ is the Gram matrix
6: Calculate the weighted summation of all loss functions $\ell^o = \mathbb{E}[\lambda^c_u \ell^c_u(\hat{S}_d, S_d) + \lambda^\phi_u \ell^\phi_u(\hat{S}_d, N_s) + \lambda^\phi_s \ell^\phi_s(\hat{S}_d, Y_o)]$
7: Train the TransNet $\phi$ w.r.t $\ell^o$ in a stochastic gradient descent manner

4 Experiment and Results

This section first describes the details of the two evaluation datasets and experiment setup. Then we give the overall experimental results to show the capability of the proposed framework to collectively prevent sensitive information leakage at once. At last, we further discuss the usability-privacy tradeoff and loss function design concerns of the proposed framework.
4.1 Datasets
To satisfy the evaluation scenario, we select datasets that satisfy the following criteria:

- The dataset should have at least two kinds of user sensitive information available for validating whether all sensitive information is successfully protected at once. This point is solely for the evaluation purpose;
- The dataset should have at least one kind of target information (e.g., human activity) available for validating whether the target information is properly unaffected;
- The dataset should have multiple subjects for validating the framework’s generalization over subjects. This point is optional but highly preferred.

We select two public inertial sensor-based human activity recognition datasets: MotionSense [11] and MobiAct [33], which meet all the above criteria. Both datasets have five kinds of user sensitive information available: gender (M/F), identification (ID), height (mm), weight (kg), and age (years old). The target information of both datasets is the activity that a user performs. Therefore, the goal of the presented framework is to disturb the inference of sensitive information, namely gender, ID, height, weight, and age, while keeping the target information, namely human activities, still being inferred successfully after data transformation.

4.1.1 MotionSense Dataset
The MotionSense dataset is collected from two inertial sensors, accelerometer and gyroscope, which are integrated within an iPhone 6s smartphone and kept in a user’s front pocket. Four sorts of time-series signals are obtained from the inertial sensors, namely attitude, rotation rate, user acceleration, and gravity. Each sort of signal has three dimensions: roll, pitch, and yaw of the attitude data and x, y, and z of the others. Thus there are 12 dimensions in the recording of each time point. A total of 24 users (10 females, 14 males) in a range of gender, age, weight, and height participate in the experiments and collect data of four daily activities: downstairs, upstairs, jogging, and walking. We remove the recordings with incomplete data or labels through data inspection, we select the data of 44 subjects (14 females, 30 males) performing four ADLs, downstairs, upstairs, walking and jogging, without data and labels of user information missing, in the form of 704 trials of 1,121,296 recordings. As there is no duration difference between the trials of the same activity, we randomly select 66% trials for training and the remaining trials for tests. Similar to the trial segmentation process of the MotionSense dataset, we cut each trial with a 50-length sliding window and obtain each sample of size (9 × 50). Finally, there are 88,412 samples for training and 22,212 samples for tests.

4.2 Experimental Setup
4.2.1 Evaluation Setup
Following the conventions in previous researches [3], [34], We use the changes of inference performance before and after transformation to validate the proposed framework. Concretely, if the accuracy of inferring human activities decreases marginally after data transformation, the proposed framework is regarded as successfully retaining target information. Otherwise, the proposed framework is regarded as failing to preserve the target information. On the other hand, if the accuracy of inferring human gender decreases considerably after data transformation, the proposed framework is regarded as successfully preventing gender information leakage. Otherwise, the proposed framework fails to protect users’ gender information. Similar evaluation criteria apply to the other sensitive information.

To validate that all the sensitive information is protected and the target information is retained after data transformation, we build six evaluation neural networks for each dataset for six kinds of information, namely human activity, gender, ID, height, weight, and age. All evaluation neural networks used the same raw bunch of data and their ground truth for training. For example, the neural network for evaluating whether the human activity information is unaffected after transformation is trained with the raw training data and human activity labels. Similarly, the evaluation network for gender is trained with raw training data and gender labels.

During the test phase, the raw test data first goes through the well-trained TransNet to achieve the transformed test data. Then the output is fed into each evaluation network respectively to get the evaluation results after the transformation. For comparison, the raw test data is fed into each evaluation network, respectively, to get the evaluation results before the transformation. The evaluation results of both before and after transformation are presented in the following Evaluation Results section.

Except for the activity classifier connected to the dense layer, the architecture of all evaluation networks is the same as that of LossNet. The evaluation networks of activity, gender, and ID use the softmax output layer for classification. For the numerical information, height, weight, and age, the linear output layer for regression analysis is used. All evaluation networks are trained using the Adam updating rule [35] with a learning rate of $10^{-3}$.

4.2.2 Training Setup
The LossNet is first trained in the trial-independent manner [11] using the raw training data and its paired human activity labels. Afterward, the parameters of the trained LossNet are fixed. Note that the LossNet is only trained with human activity labels since this information is non-sensitive and can be public. To achieve the “style” loss $\ell^2_N$ for training the TransNet, we generate the noise random noise $N$ from random numbers of a uniform distribution.
The HAR accuracy can still maintain above 90% with only less than a 4% drop. The HAR accuracy of the MobiAct dataset has a relatively larger drop than the MotionSense dataset (~ 4% vs. ~ 1%). It is noticeable that this performance can be optimized by tuning the loss weights \( \lambda_i \) (more in the Privacy-Usability Tradeoff section), and the settings of the loss weights for evaluating both datasets are identical so that the reported results are not optimal separately. This demonstrates that our framework is robust to the settings of loss weights across different datasets. In contrast, when inferring user gender, the accuracy declines dramatically nearly to the random guess level. Note that due to the gender imbalance of the evaluation datasets, the random guess level of gender inference is 58% and 68% for the MotionSense and MobiAct dataset, respectively. There is also a considerable drop of ID inference accuracy after data transformation with the relative result decrease of around 85% for both datasets. Thus the sensitive information of user gender and ID has been changed to have a random “style” and hard to be precisely inferred after data transformation. The inference errors of numerical sensitive information, height, weight, and age, also rise remarkably after data transformation. In particular, the inference of user weight experiences the most significant performance degradation with relative MSE increases 46.06 times and 21.36 times after transformation for the MotionSense and MobiAct datasets. Even the smallest performance degradation of inferring user age still suffers more than five times MSE increases. Considering the user age has a relatively small reasonable range, the increase of the inference error is significant. The overall results exhibit that our framework is able to transform raw mobile sensor signals into a new representation that has a “style” (sensitive information) similar to random noise, yet the “content” (target information) same with raw data.

### 4.3 Evaluation Results

#### 4.3.1 Overall Performance

Figure 3 plots the evaluation results of the proposed framework on two evaluation datasets. The numerical details are summarised in Table 2. The upper part of Figure 3 displays the absolute results of inferring both target and sensitive information. We use the classification accuracy as the evaluation criteria for inferring categorical information, namely activity, gender, and ID, and plot the results with a normal scale to the left axis of Figure 3a and 3b. For continuous information (i.e., height, weight, and age), we introduce the mean squared error (MSE) as the evaluation criteria and plot the results with a log scale to the right axis of Figure 3a and 3b. The bottom part of Figure 3 exhibits the relative results of the transformed data compared to the raw data. The relative result is defined as:

\[
\Delta_r = \frac{R(\text{After}) - R(\text{Before})}{R(\text{Before})},
\]

where \( R(\text{After}) \) and \( R(\text{Before}) \) are the absolute results of transformed data and raw data. As the sensitive information has real-world reasonable ranges, the absolute results give the intuitive sense about the extent that the proposed framework perturbs the sensitive information. On the contrary, the relative results depict the change extent after data transformation compared to raw data.

Our framework obtains satisfactory performance of preventing sensitive information leakage on both datasets with a marginal decrease of HAR accuracy, but significant error increases in inferring all sensitive information. Specifically, after data transformation, the HAR accuracy can still maintain above 90% with only less than a 4% drop. The HAR accuracy of the MobiAct dataset has a relatively larger drop than the MotionSense dataset (~ 4% vs. ~ 1%). It is noticeable that this performance can be optimized by tuning the loss weights \( \lambda_i \) (more in the Privacy-Usability Tradeoff section), and the settings of the loss weights for evaluating both datasets are identical so that the reported results are not optimal separately. This demonstrates that our framework is robust to the settings of loss weights across different datasets. In contrast, when inferring user gender, the accuracy declines dramatically nearly to the random guess level. Note that due to the gender imbalance of the evaluation datasets, the random guess level of gender inference is 58% and 68% for the MotionSense and MobiAct dataset, respectively. There is also a considerable drop of ID inference accuracy after data transformation with the relative result decrease of around 85% for both datasets. Thus the sensitive information of user gender and ID has been changed to have a random “style” and hard to be precisely inferred after data transformation. The inference errors of numerical sensitive information, height, weight, and age, also rise remarkably after data transformation. In particular, the inference of user weight experiences the most significant performance degradation with relative MSE increases 46.06 times and 21.36 times after transformation for the MotionSense and MobiAct datasets. Even the smallest performance degradation of inferring user age still suffers more than five times MSE increases. Considering the user age has a relatively small reasonable range, the increase of the inference error is significant. The overall results exhibit that our framework is able to transform raw mobile sensor signals into a new representation that has a “style” (sensitive information) similar to random noise, yet the “content” (target information) same with raw data.

#### 4.3.2 Privacy-Usability Tradeoff

The loss weight \( \lambda_i \) controls the tradeoff between privacy protection and the usability of transformed data. In this section, we perform experiments by varying the weight of the style loss \( \lambda_o \) from 0.05 to 0.95 to investigate the privacy-usability tradeoff of the proposed framework. The content loss and the usability loss change correspondingly with the style loss and equally share the change value of the style loss to make the overall summation kept to constant. Figure 4 shows the results of both the MotionSense and MobiAct datasets. The upper part shows the absolute value results, and the bottom part displays the relative results.

It is evident that with the weight of the style loss growing, the inference error of all sensitive information increases; thus, the risk of sensitive leakage decreases. However, the accuracy of human activity recognition does not change too much until the weight of style loss higher than 0.85. Especially for the MotionSense dataset, the activity recognition accuracy remains about 90% at the style loss weight of 0.85. At the lower side, the MSE and inference accuracy of the sensitive information change obviously at 0.25 and fluctuate smoothly afterward. There is a sharp change after the style loss larger than 0.85. Similarly, the activity recognition accuracy drops clearly at the tail part, which indicates that the usability of data decreases significantly at a large style loss weight. The MobiAct dataset has a similar privacy-usability tradeoff trend as the MotionSense dataset with sensitive information protection efficiency goes up apparently after the style weight of 0.45. However, the activity recognition accuracy drops a noticeable amount of about 15% with the style weight from 0.65 to 0.85. Finally, the activity inference accuracy sharply down to
only 20% at the end, where the transformed data is useless. Thus, it is crucial to carefully select the weight of the style loss to keep a satisfactory privacy-usability tradeoff.

**4.3.3 Visualization**

To gain an intuitive sense of the transformation process, we compare the spectrograms of acceleration signals before and after transformation. Figure 5 shows the visualization results. We can observe that the transformation process introduces new periodic components that cover the original ones and differ across activities. Since the periodic information of the motion sensor data encodes abundant signatures of users, the perturbation on such components reduces the possibility of sensitive information leakage. The downstairs and upstairs visualization shows similar perturbation patterns because people act slightly differently when going upstairs and downstairs concerning the acceleration of the X-axis. In contrast, the visualization patterns of walking and jogging are significantly different, and both different from that of upstairs and downstairs.

**4.3.4 Impact of Loss Functions**

In this section, we discuss the impact of different loss functions $\ell^i_\phi$ on the performance of the presented framework. A reconstruction loss defined as $\ell^r_i = \frac{1}{md} \|S_d - \hat{S}_d\|_2^2$ that encourages the transformed data to be the same with the raw data is also introduced for a comparison study as this loss function is commonly used in previous studies for sensitivity-protecting data transformation [11].

Figure 6 (a)-(c) show the sensitivity protection performance of the proposed framework trained solely using an individual loss function on the MotionSense dataset. When training only with the usability loss $\ell^u_\phi$, as shown in Figure 6(a), the transformed data can keep approximately the same usability characteristics as the raw data. However, even though the transformation process has modified the raw data that the transformed data and the raw data have different distributions, the sensitive information is still retained unintentionally after data transformation. This result is in consonance with that reported in [3]. Figure 6(b) reveals the evaluation results achieved only with the content loss $\ell^c_\phi$. The $\ell^c_\phi$ promotes more information preservation than the usability loss $\ell^u_\phi$ since $\ell^c_\phi$ encourages the extracted features of the transformed data to be the same with the raw data rather than only the final recognition accuracy. Besides, the content loss can be used when the ground truth of the target information is unavailable, and the LossNet is a general pretrained neural network that is not specifically designed for inferring the target information. In contrast, as indicated in Figure 6(c), the style loss $\ell^s_\phi$ advocates all information damaged to a massive extent that the activity recognition accuracy drops to a random guess level.

In Figure 6(d), although the reconstruction loss helps to retain more raw information even including some hidden ones than the other loss functions, it is such a tight constraint that degrades the sensitive protection considerably. On the contrary, the content loss is a relatively loose constraint compared to the reconstruction loss, yet also has potential abilities to preserve unknown information. Thus we use the content loss rather than the reconstruction loss, which is often used in previously reported transformation-based sensitive information protection strategies.
Figure 6. Impact of Different Loss Functions. The activity, gender, and ID are evaluated with classification accuracy; the height, weight, and age are evaluated with MSE.

5 CONCLUSION

In this paper, we propose a novel data transformation framework for collectively preventing sensitive leakage through mobile sensor signals. The presented work transforms raw sensor signals into new representations with a “content” same as the raw data but with a “style” like random noise. Unlike previous works that can protect only one specific information at once and require the ground truth of sensitive information for training, our framework can protect all sensitive information at once and does not require the labels of sensitive information from users. We evaluate the proposed framework on two multi-sensor human activity datasets for protecting all five sorts of user sensitive information, namely gender, ID, height, weight, and age. The results demonstrate that our framework can protect all user sensitive information at once through the random “style” transformation and retain the target information with a marginal recognition accuracy drop.

REFERENCES


Kaixuan Chen is an Assistant Professor at the Department of Computer Science, Aalborg University, Denmark. Before joining Aalborg University, she received her PhD from the School of Computer Science and Engineering, University of New South Wales (UNSW). She earned her Bachelor’s degree from Xi’an Jiaotong University (XJTU) in 2015, majoring in Telecommunication Engineering. Currently, her scientific research interest is in Data Mining, Deep Learning and Internet of Things (IoT).

Zheng Yang received the BE degree in computer science from Tsinghua University, in 2006 and the PhD degree in computer science from the Hong Kong University of Science and Technology, in 2010. He is currently an associate professor at the Institute of Trustworthy Network and System, School of Software, Tsinghua University. His main research interests include wireless ad-hoc/sensor networks, and mobile computing. He is a member of the IEEE and the ACM.

Xin Gao received the B.S. degree in computer science from Tsinghua University, Beijing, China, in 2004, and the Ph.D. degree in computer science from University of Waterloo, Waterloo, ON, Canada, in 2009. He is currently an Associate Professor of computer science with the Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia. He is also a PI with the Computational Bioscience Research Center, KAUST, and an Adjunct Faculty Member with the David R. Cheriton School of Computer Science, University of Waterloo. His group focuses on building computational models, developing machine learning methods, and designing efficient and effective algorithms, with particular a focus on applications to key open problems in biology. He has co-authored more than 100 research articles in the fields of machine learning and bioinformatics.

Yunhao Liu received the B.S. degree from the Automation Department, Tsinghua University, Beijing, China, the M.A. degree from Beijing Foreign Studies University, Beijing, and the M.S. and Ph.D. degrees in computer science and engineering from Michigan State University (MSU), East Lansing, MI, USA. He is currently an MSU Foundation Professor and the Chairperson of the Department of Computer Science and Engineering, Michigan State University, and holds a Chang Jiang Chair Professorship with Tsinghua University. His current research interests include sensor network and pervasive computing, peer-to-peer computing, Internet of Things, and supply chain. Dr. Liu currently serves as the Editor-in-Chief for ACM Transactions on Sensor Networks. He was an Associate Editor of the IEEE/ACM TRANSACTIONS ON NETWORKING from 2012 to 2016, and an Associate Editor-in-Chief of the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS from 2011 to 2015. He is a fellow of the IEEE and the ACM.

Dalin Zhang is currently an Assistant Professor at the Department of Computer Science, Aalborg University, Denmark. Before joining Aalborg University, he received his PhD degree at the University of New South Wales Sydney (UNSW Sydney), Australia in 2020. Before pursuing his PhD degree, he was at Spreadtrum Communications, Inc. China, working as a Digital Integrated Circuit Design Engineer. He received his Master degree from University of Chinese Academy of Sciences in 2015, and my Bachelor Degree from Jilin University in 2012, both major in Microelectronics. His research interest includes brain-computer interface (BCI), human activity recognition and Internet of Things (IoT).

Lina Yao is currently an Associate Professor at School of Computer Science and Engineering, the University of New South Wales (UNSW), Australia. She received her PhD degree and Master degree both from the University of Adelaide (UoA) in 2014 and 2010, respectively, and her Bachelor degree from Shandong University (SDU). She was ARC Discovery Early Career Researcher Award (DECRA) Fellow between 2016-2018 (awarded in 2015). Before she joined UNSW in 2016, she was a lecturer and an ARC research associate at UoA. Her research interest lies in Data Mining and Machine Learning applications with the focuses on Internet of Things Analytics, recommender systems, human activity recognition and Brain Computer Interface. She is a member of the IEEE and the ACM.