Ensemble classifier of Long Short-Term Memory with Fuzzy Temporal Windows on binary sensors for Activity Recognition

Javier Medina-Quero[a,*], Shuai Zhang[b], Chris Nugent[b], M. Espinilla[a]

[a] *Department of Computer Science, University of Jaén, Campus Las Lagunillas, Jaén 23071, Spain*
[b] *School of Computing, Ulster University, Co. Antrim, Northern Ireland BT37 0QB, United Kingdom*

Abstract

There are approaches that successfully recognize activities of daily living by using a trained classifier on feature vectors created from binary sensor data. Although these approaches have been successful, there are still open issues such as the evaluation of multiple temporal windows, ensembles of classifiers or unbalanced classes which need to be addressed in order to improve the performance of the real-time activity recognition process. In this paper, we present a methodology for Real-Time Activity Recognition based on the diverse fields of Machine Learning, including Fuzzy Logic and Recurrent Neural Networks. The methodology uses a long-term and short-term representation of binary-sensor activations based on Fuzzy Temporal Windows. The paper proposes an ensemble of activity-based classifiers for the purposes of balanced training, where each classifier in the ensemble is a Long Short-Term Memory. The approach was evaluated using two binary-sensor datasets of daily living activities and benchmarked against previous approaches based on the combination of sensor activation features.

Keywords: Activity Recognition, Fuzzy Temporal Windows, Long Short-Term Memory, Unbalanced Data, Ensemble architectures.

1. Introduction

Activity recognition (AR) systems deployed in smart homes are characterized by their ability to detect human actions and their goals in order to improve assistance. Such assistive technologies have started to be adopted by smart homes and healthcare applications in practice and have delivered promising results for improving the quality of care services for the ageing and

*Corresponding author. Address: Department of Computer Science, University of Jaén, Campus Las Lagunillas, Jaén 23071, Spain. Tel: +34 953 212897

Email addresses: `jmquero@ujaen.es` (Javier Medina-Quero), `s.zhang@ulster.ac.uk` (Shuai Zhang), `cd.nugent@ulster.ac.uk` (Chris Nugent), `mestevez@ujaen.es` (M. Espinilla)

provision of responsive assistance in emergency situations (Chen, Hoey, Nugent, Cook & Yu, 2012).

AR based on the use of binary sensors is a useful approach to assess the status of daily living within a sensorised environment in an unobtrusive manner (Espinilla, Medina, Calzada, Liu, Martinez & Nugent, 2017b; Krüger, Nyolt, Yordanova, Hein & Kirste, 2014). Binary sensors are small and light devices which installed in everyday objects to register human interaction. Examples of these kinds of devices are motion detectors, contact switches, break-beam sensors, and pressure mats (Wilson & Atkeson, 2005). They are easily connected to a middleware, generally using wireless communications, and subsequently generate streams of binary data.

Data Driven Approaches for AR, which aim to use the information gleaned from the sensors, require a large dataset with the activities labelled specifying their starting and ending point in time to represent the ground truth. The activities within such a dataset set are usually performed by participants as part of a controlled experiment whilst the binary sensors are activated in controlled conditions within a smart lab/home. A training process is necessary for data driven approaches to build an activity model. Once trained the model can be exposed to unseen data to evaluate its generalization abilities in classifying unseen activities (Gu, Wang, Wu, Tao & Lu, 2011; San Martín, Peláez, González, Campos & Lobato, 2010; Espinilla, Liu & Chamizo, 2017a).

A number of previously undertaken AR studies have been centered on classifying activities, which were previously labelled by human-defined time intervals (Espinilla, Rivera, Pérez-Godoy, Medina, Martinez & Nugent, 2016). These approaches are referred to as *explicit segmentation* (Krishnan & Cook, 2014) and do not provide real-time capabilities in AR. Within a real context, however, it is desirable for any AR model to be capable of running in real-time (Cook, Schmitter-Edgecombe, Crandall, Sanders & Thomas, 2009). Real-time refers to the recognition of activities: i) when they are being undertaken (Yan, Liao, Feng & Liu, 2016), ii) while new sensor events are being recorded, and iii) the processing of data to produce results within an acceptable period of time (Martin, 1965). Including real-time capabilities has become a key challenge in AR to provide responses to real-world conditions (Chen et al., 2012) enabling *adequate assistance services*, which can be offered within Ambient Assisted Living (Storf, Kleinberger, Becker, Schmitt, Bomarius & Prueckner, 2009). In real-time AR, explicit information relating to the labelled time interval is generally not included whilst learning. This subsequently requires window based approaches to segment the data stream (Krishnan & Cook, 2014).

The main difficulty with real-time AR approaches is the ability to correctly define the size of the temporal window to allow effective recognition of activities (Ordónez, de Toledo & Sanchis, 2013; Banos, Galvez, Damas, Pomares & Rojas, 2014). The difficulty of using a single sliding window is that the more sensor events from the past that are included, implying noise in the trained model (Espinilla, Medina, Hallberg & Nugent, 2018; Banos, Galvez, Damas, Pomares & Rojas, 2014; Krishnan & Cook, 2014). In this work, the use of multiple temporal windows and fuzzy aggregation methods are proposed to enable the long and mid term evaluation of sensors, discriminating sensor activations by more than one temporal window.

A further issue when considering the development of Data Driven approaches to AR is that the datasets used usually suffer from a class imbalance problem (Van Kasteren, Noulas, Englebienne & Kröse, 2008), (Ordónez et al., 2013), where activity events are extremely scarce (Yin, Yang & Pan, 2008). For example, *cooking* is an activity which usually has a duration around one hour with a many sensor activations, while *go to the toilet* typically only lasts a few minutes with a few sensor activations. Balancing methods are recommended to assist with improving generalization of the model (Logan, Healey, Philipose, Tapia & Intille, 2007). An example of balancing method is to learn using data leads to minority classes, ignoring majority classes, when a specific classifier for a minority class is trained (Guo, Yin, Dong, Yang & Zhou, 2008).

This work takes these issues into consideration proposing a Data Driven Approach whose methodology aims:

- to propose a representation focused on the long-term and short-term based on a temporal sequence, which is defined by multiple and incremental fuzzy temporal windows under a fuzzy aggregation. Here, shorter term is related to finer temporal granularity and the longer term is related to coarser granularity. Together they provide an adequate aggregation of past events whilst at the same time provide an accurate representation of recent events from binary data streams.

- to develop a more representative and balanced approach to learning in order to: i) increase the learning capabilities using an ensemble of classifiers, and ii) create a training dataset based on the similarity relation between activities, which encourages learning in conflicting activities.

The proposed approach is introduced in Section 2 along with related works. The proposed approach is formally defined in Section 3. The proposed methodology is evaluated on two popular datasets, (Ordónez et al., 2013) and (Singla et al., 2009) in Section 4. Finally, in

Section 5, conclusions and ongoing work are discussed.

2. Related works

The proposed approach presented in this work is based on two main concepts: i) a representation based on fuzzy temporal windows to describe long-to-short sequences of binary-sensor activations suitable for sequence classifiers and, ii) learning an ensemble of classifiers based on activity similarity to address the class imbalance problem.

2.1. Representation of temporal data for real-time Activity Recognition

AR is an open field of research where approaches based on different types of sensors have been proposed. Wearable devices have been used to analyze activities such as walking, sitting or lying down (Reyes-Ortiz, Anguita, Ghio & Parra, 2012; Ortiz, 2015; Medina, Fernandez-Olmo, Peláez & Espinilla, 2017b). Approaches based on body-worn devices can, however, be intrusive for daily life (Roggen, Calatroni, Rossi, Holleczek, Förster, Tröster, Lukowicz, Bannach, Pirkl, Ferscha et al., 2010) and, at the same time, they are not appropiate to provide a long-term vision of daily activities due to being focused on gesture rather than a user's interaction with the smart environment.

Vision based sensors have also been used as a rich source for the recognition of human activities, generally in outdoor environments (Robertson & Reid, 2006). Within the home-based literature description of activities through the detection of human joints with a vision sensor has been considered (Rege, Mehra, Vann & Luo, 2017). In addition, the wearable vision sensors (Shewell, Medina-Quero, Espinilla, Nugent, Donnelly & Wang, 2017) in daily object interaction has been reported. The main disadvantages of vision sensor approaches include the high computational costs, in addition to privacy concerns (Sixsmith & Johnson, 2004).

Binary sensors have been proposed as suitable devices for describing daily human activities within a smart environment setting (Van Kasteren et al., 2008). Their main advantages are that they are: i) easy to install, ii) small in size, iii) low-cost and iv) minimally invasive in comparison to videos and microphones (Tapia, Intille & Larson, 2004). Their main drawback, however, is their ability to manage situations of multi-occupancy within smart environments. More recently, approaches based on binary sensors have been extended to Smart Meters (Koutitas & Tassiulas, 2016), which enable identifying user interaction with electrical devices (Belley, Gaboury, Bouchard & Bouzouane, 2013) based on power consumption (Alcalá, Ureña, Hernández & Gualda, 2017).

Within the domain of real-time AR with binary sensors, the definition of a temporal window size requires a deep analysis to segment the data correctly (Banos et al., 2014; Espinilla et al., 2018). To date, several attempts have evaluated the performance of Machine Learning classifiers, such as, Decision Trees (DT), Support Vector Machines (SVM), naives Bayes (NB) or Hidden Markov Model (HMM), based on sliding windows (Stikic, Huynh, Van Laerhoven & Schiele, 2008; Tapia, Intille & Larson, 2004; Yala, Fergani & Fleury, 2015) or dynamic windows (Yan et al., 2016; Krishnan & Cook, 2014; Espinilla et al., 2018).

These previous efforts highlight studies which have strived to adjust fixed or dynamic window sizes in real-time AR involving context-based complexity (Shahi, Woodford & Lin, 2017), which we aim to avoid.

From an evaluation perspective, real-time AR has mainly been provided by two methods:

- usage of time-slots: In this approach, the process of recognizing the activity is evaluated for each given time-slot with a given duration (Van Kasteren, Englebienne & Kröse, 2010). Related approaches have found that a time-slot of 60 seconds provides good performance in human activity recognition with binary sensors (Van Kasteren et al., 2010; Ordónez et al., 2013). An adaption of this approach has been used for the purposes of evaluation within the current works.

- usage of events: In this approach, the sensor data stream is evaluated taking into account the changes in sensor events (Patterson et al., 2017). With this method, the approach classifies every sensor event based on the information encoded in a sliding window of preceding sensor events. This approach is mainly adopted by research studies that i) analyze sensors which provide continuous data from wearable devices, such as accelerometers (Banos et al., 2014), and ii) using binary sensors in AR together with dynamic windowing approaches (Shahi et al., 2017).

2.2. Fuzzy temporal windows to describe long-to-short sequences of binary-sensor activation

In this work, a methodology is proposed to aggregate data from binary sensors by means of incremental temporal windows to recognize daily activities in real-time from a single occupancy scenario. In previous works, a combination of human-defined features in binary sensors, such as last activation, change point or raw activation in current time, has been proposed as a suitable representation for features in real-time AR (Van Kasteren et al., 2010) under windowing approaches (Huynh, Blanke & Schiele, 2007). The use of dynamic windowing and human-defined features for representing binary sensors has been evaluated previously considered (Shahi et al., 2017). Nevertheless, these representations based on human interpretation lack a longer temporal representation, which has been recognized as a critical aspect impinging upon the performance of sliding window approaches (Espinilla et al., 2018).

Taking these previous findings into consideration, the use of long-term or mid-term windows is considered a key to represent events from the previous minutes or hours, which can be critical in the overall recognition of a given activity. For example, the difference between breakfast and dinner is usually provided by activations as well as deactivations of the bed sensor from long-term to short-term. In the case of breakfast, the short-term deactivation of the bed sensor. While dinner, the bed sensor is deactivated from the longer term to short term. Therefore, there is a relation between the sleep/sitting in the bed with the breakfast/dinner activity.

In order to define the activation for each sensor within a long-term, mid-term and short-term representation, this work proposes the definition of Fuzzy Temporal Windows (FTWs) of incremental size, which represent the temporal activation from binary-sensors. With such an approach, the selection of a critical single window size is reduced by offering a wide range of FTWs (refer to Figure 1). In developing intelligent systems from sensor data stream, Fuzzy Logic (Zadeh, 1996) has provided successful results in aggregating sensor information using linguistic representations (Medina, Espinilla, Zafra, Martínez & Nugent, 2017a; Espinilla & Nugent, 2017). The proposed FTWs (Medina, Martinez & Espinilla, 2017c) provide a model to: i) weight sensor activation based on temporal membership functions, ii) define progressive and interpretable temporal windows, and iii) reduce the complexity of the temporal representation from long and mid-term sensor activation.

Furthermore, the use of FTWs can define a temporal-sequence representation of sensor activation by means of incremental temporal size (Edwards, Coward, Hamer, Twitchen & Hobson, 2000; Rossetti, Milli, Giannotti & Pedreschi, 2017). The advantage of including a sequence representation is the improvement in learning by a sequence classifier, such as Long Short-Term Memories (LSTM) (Hochreiter & Schmidhuber, 1997). LSTM is a Recurrent Neural Network which is formed by a chain of repeated modules called memory cells. Each memory cell is composed of an input gate, a self-recurrent connection, a forget gate and an output gate (refer to Figure 1). The cell states of LSTMs can be controlled in order to remove or add information based on the learning of the gates. LSTMs have also provided state-of-the-art learning of video representations (Srivastava et al., 2015), which have been adapted to AR from fixed windows or video shot segments of daily activities (Donahue, Anne Hendricks, Guadarrama, Rohrbach, Venugopalan, Saenko & Darrell, 2015).

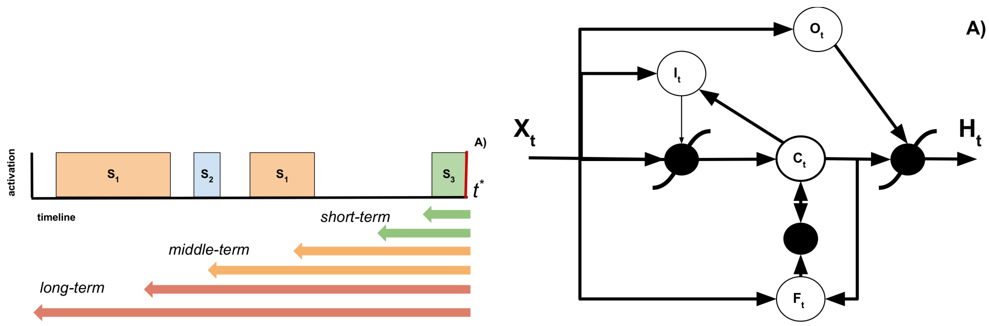

Figure 1: A) Relevance of activation for some sensors (S_1, S_2, S_3) in a given current time t^* within a long-term, mid-term and short-term representation. B) Scheme of a LSTM cell.

LSTMs have been previously described as an appropriate choice for use in sensor-based AR: i) (Singh et al., 2017), where together with Convolutional Neural Networks (CNN), they have been considered under a time slice approach; and ii) (Ordóñez & Roggen, 2016), where an approach based on wearable devices, human activity recognition and human gesture recognition was presented. In both instances, LSTM has provided high accuracy in learning activities, however, avoided aggregation of long and mid-term representations. For example, (Singh et al., 2017) proposed 70 sequences for one-minute time-slots, and (Ordóñez & Roggen, 2016) used a 500 ms sliding window with 250 ms time-slots (we note the difference between the binary and wearable sensor representation of time). In this work, we propose the use of FTWs to describe multiple temporal windows, whose size is defined from a half day to a few minutes by means of a long-to-short term fuzzy temporal representation.

In Tables 1 and 2, we summarize the related work considered for different types of sensors in AR and the representation of features and classifiers, respectively.

Table 1: Example approaches with different types of sensors in AR

Type	Reference works	Advantages	Disadvantages
Vision	(Rege et al., 2017; Robertson & Reid, 2006; Shewell et al., 2017; Sixsmith & Johnson, 2004; Donahue et al., 2015)	Low invasiveness, rich visual description	Privacy, cost, computational burden
Wearable	(Ortiz, 2015; Ordóñez & Roggen, 2016; Banos et al., 2014)	Individual granularity, precision, high collecting rate	Invasive, limited to gesture recognition, computational burden
Binary	(Van Kasteren et al., 2010; Ordónez et al., 2013; Singh et al., 2017; Yan et al., 2016; Krishnan & Cook, 2014; Yala et al., 2015; Espinilla et al., 2018)	Easy-to-install and maintain, minimally invasive, low-cost	Non-individual granularity, low performance in multi-occupancy
Sensing Electricity Data	(Belley et al., 2013; Alcalá et al., 2017)	Easy-to-install, minimally invasive	Non-individual granularity, low performance in multi-occupancy

Table 2: Representation of features and classifiers of AR

Reference works	Representation	Classifier
(Yan et al., 2016; Krishnan & Cook, 2014; Espinilla et al., 2018)	Windowing approaches (binary sensors)	NB+SVM+HMM+Others
(Ortiz, 2015; Banos et al., 2014)	Windowing approaches (wearable sensors)	SVM+Decision Trees+Bayes+Others
(Van Kasteren et al., 2010; Ordónez et al., 2013)	Human-defined features+ short sliding window for time-slots	DT+SVM+HMM+Others
(Shahi et al., 2017)	Human-defined features+ dynamic window	NB
(Donahue et al., 2015)	CNN+ Fixed sequences (video sensors)	LSTM
(Singh et al., 2017)	CNN+fixed sequence of time-slots (binary sensors)	LSTM
(Ordóñez & Roggen, 2016)	CNN+Sliding window (wearable sensors)	LSTM

2.3. *Balanced-based similarity training for an ensemble of classifiers*

A dataset is considered to be imbalanced when its classes are not equally represented (Chawla, Bowyer, Hall & Kegelmeyer, 2002; Japkowicz & Stephen, 2002). It has been previously reported that daily activity datasets suffer from a severe class imbalance problem (Ordónez et al., 2013; Van Kasteren et al., 2008). In Neural Network (NN) approaches, the classifier performance deteriorates with even the most modest of class imbalance in the training data (Mazurowski, Habas, Zurada, Lo, Baker & Tourassi, 2008).

Due to our approach being based on sequence learning through NN (specifically LSTM) from imbalanced daily activity datasets, we propose the following two key points to minimize the impact of the imbalance problem.

Firstly, an ensemble of activity-based classifiers is included. Ensemble of classifiers have been proposed to handle class imbalance problems (Galar, Fernandez, Barrenechea, Bustince & Herrera, 2012). The use of an ensemble of classifiers for wearable-based activity recognition has been previously proposed in (Lester, Choudhury, Kern, Borriello & Hannaford, 2005). In this approach, a feature selection together with an ensemble of static classifiers and hidden Markov models were proposed to recognize different activities. In (Catal, Tufekci, Pirmit & Kocabag, 2015) an ensemble of machine learning classifiers werw evaluated for accelerometer-based AR.

Secondly, a balancing method has been included to introduce an *ad hoc* training dataset for each specific activity. The balancing method is developed by a random process which weights the ratio of samples according to the similarity between activities. Similar random sampling has been described as an effective method for dataset balancing (Sanchez, Martinez & Gonzalez, 2017; Möller-Acuña, Ahumada-García & Reyes-Suárez, 2017), having been included in common tools, such as, Weka and R. In addition, balancing the data based on similarity has been proved to deal with the problem of learning from imbalanced datasets (Batista, Prati & Monard, 2004). In this work, we have computed the similarity based on the statistical analysis of the activation of sensors within the activation of activities as it provides a useful metric when evaluating AR (Carnevali, Nugent, Patara & Vicario, 2015).

The combination of both methods, which weight the training dataset based on the conflict between classes within an ensemble of classifiers, has been demonstrated to create strong learning methods (Dietterich, 2000).

In Section 3, we detail the methodology for i:) defining long-to-short sequences in binary-sensor activation using fuzzy temporal windows and ii) balancing the training in ensemble learning classifiers based on activity similarity.

3. Methodology

In this Section, the proposed methodology is presented for defining the activation of binary sensors and activities in different ranges of time.

A set of binary sensors is represented by $S = \{S_1, \ldots, S_{|S|}\}$ and a set of daily activities is represented by $A = \{A_1, \ldots, A_{|A|}\}$, where $|S|$ and $|A|$ are the number of sensors and daily activities respectively. Each of the binary sensors and daily activities are described by a set of binary activations within a set of ranges of time, which are defined by a starting and ending point in time as shown by Eq. (1):

$$\begin{aligned} S_i &= \{S_{i_0}, \ldots, S_{|S_i|}\}, S_{i_j} = \{S^0_{i_j}, S^+_{i_j}\} \\ A_i &= \{A_{i_0}, \ldots, A_{|A_i|}\}, A_{i_j} = \{A^0_{i_j}, A^+_{i_j}\} \end{aligned} \tag{1}$$

being i) $|S_i|$, $|A_i|$ the total number of activations for a given binary sensor S_i and a daily activity respectively, and ii) $S^0_{i_j}$,$S^+_{i_j}$ the starting and ending point of a given time of activation respectively.

3.1. From activation between ranges of time to segmented time-slots

The first step in processing the dataset is to generate a segmented timeline of time-slots (van Kasteren et al., 2011). Given that the activation of sensors and activities is defined by ranges of time, we are able to generate the segmented timeline which indicates the activation of activities and sensors for a given time interval Δt.

Subsequently, we divide the timeline $T = \{min(S^0_{i_j}), max(S^+_{i_j})\}$, which configures the range of time between minimal starting point and maximal ending point, in time-slots t_i of equal duration Δt. The range of evaluation for each time-slot is defined by a sliding window between $[t_i, t_i + \Delta t]$. For each time-slot and a given sensor, we determine its activation based on whether it has been activated (even just partially) within it:

$$S(t_i, s) = \begin{cases} 1 & \exists [S^0_{s_j}, S^+_{s_j}] \cap [t_i, t_i + \Delta t] \forall S_{s_j} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

In a similar way, for each time-slot and a given activity, we determine its activation based on whether it has been carried out (even just partially) within it:

$$S(t_i, a) = \begin{cases} 1 & \exists [A^0_{i_j}, A^+_{i_j}] \cap [t_i, t_i + \Delta t] \forall A_{i_j} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

In this way, the segmented timeline can be represented as a binary matrix where for each time-slot t_i we return the value 1 if a sensor or an activity has been active. Additionally, the segmented timeline for a given sensor or activity is represented as a row which involves the activation in all time-slots $S(s) = \{S(t_0, s), \ldots, S(t_n, s)\}$. For the sake of simplicity, we refer to t^+ as a time-slot t_i in the timeline T.

An example of segmentation from temporal activation ranges to time-slots is presented in Figure 2.

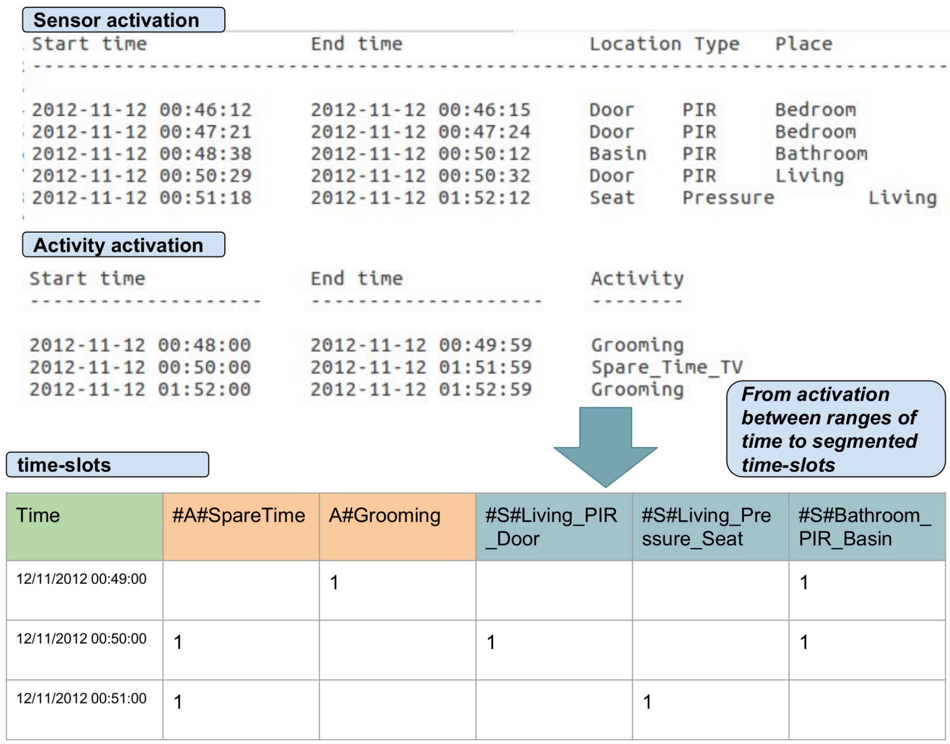

Figure 2: Example of the segmentation for activity and sensor activation ($\Delta t = 60s$). The labels beginning with $\#A$ represent an activity and $\#S$ represent a sensor.

3.2. Representation of sensor activation based on Fuzzy temporal windows

In this Section, a binary-sensor representation approach is described, which is based on fuzzy temporal windows (FTWs). A real-time classification of daily activities is proposed to decide which activity, or the absence thereof (*Idle*), is identified for all time-slots in the timeline.

Based on previous works (Medina, Espinilla & Nugent, 2016),(Medina et al., 2017c), a fuzzy aggregation of the sensor activation has been integrated within the segmented timeline using FTWs. In this previous work, a fuzzy temporal aggregation of wearable and binary sensors was proposed to define an interpretable representation of a smart environment.

FTWs are therefore described from a given current time t^* to a past point in time t_i as a function of the temporal distance $\Delta t_i^* = t^* - t_i, t^* > t_i$ (Medina et al., 2016). For this purpose, a given FTW T_k relates the sensor activation $S(s, t_i)$ in a current time t^* to a fuzzy set $T_k(\Delta t_i^*)$, which is characterized by a membership function $\mu_{\tilde{T}_k}(\Delta t_i^* = t^* - t_i)$. A given FTW can be represented as $T_k(\Delta t_i^*)$ instead of $\mu_{\tilde{T}_k}(\Delta t_i^*)$.

For a given FTW T_k and the current time t^*, each past sensor activation $S(t_i, s)$ is weighted by calculating the degree of time-activation within the fuzzy temporal window T_k according to

Eq. (4).

$$T_k(s, t^*, t_i) = S(t_i, s) \cap T_k(\Delta t_i^*), t_i <= t^* \tag{4}$$

The degrees of time-activation are subsequently aggregated using the t-conorm operator in order to obtain a single activation degree of both fuzzy sets $S(s) \cap T_k$ by Eq. (5).

$$T_k(s, t^*) = S(s) \cup T_k(\Delta t^*) = \bigcup_{\bar{t_i} \in T} S(t_i, s) \cap T_k(\Delta t_i^*), t_i <= t^* \tag{5}$$

We note several fuzzy operators can be applied to implement semantics in the t-norms. Nevertheless. In this paper we propose the use of maximal and minimal operators, being $\bigcup = max, \cap = min$, which compute the highest degree of activation within the fuzzy temporal window. They are recommended for representing binary sensors due to they present a low activation rate (Medina et al., 2017c).

$$T_k(s, t^*) = S(s) \cup T_k(\Delta t^*) = max(min(S(t_i, s), T_k(\Delta t_i^*)), \forall t_i \in T, t_i <= t^* \tag{6}$$

An example of the representation of a FTW within a sensor activation in the segmented timeline is presented in Figure 3.

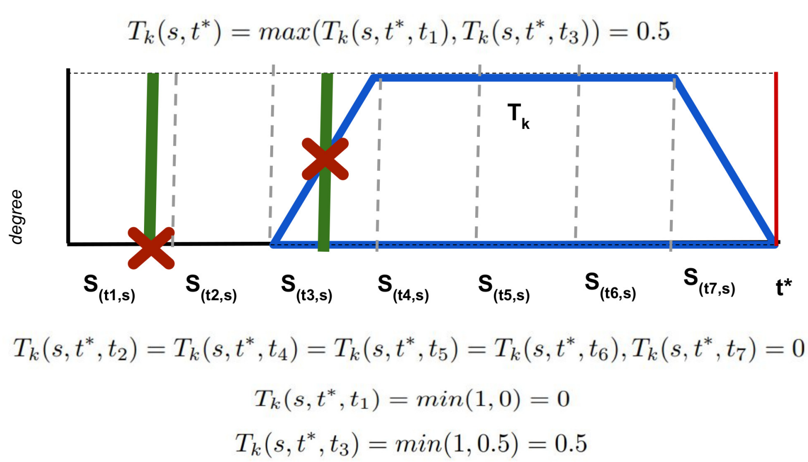

Figure 3: Representation of a fuzzy temporal window T_k within a sensor activation in the segmented timeline $S(s, t_i)$. In the example, the aggregated degrees of activation $T_k(s, t^*)$ are 0.5.

3.3. LSTM for sequence classification of FTW

The representation of a sensor activation based on FTWs can be used to define a sequence for the purposes of classification. In this work, we propose to define multiple FTWs with incremental temporal size. The aim of this representation is to collect long-term to short-term sensor activations, where shorter activations have finer temporal granularity and the longer activations have a coarser temporal granularity.

A simple definition of FTWs is described, the durations of which are represented by a fuzzy set characterized by a membership function whose shape corresponds to a trapezoidal function. Other membership functions can also be used to define FTWs, however, the trapezoidal shape is proposed to define the limit in a straightforward way. The well-known trapezoidal membership functions are defined by a lower limit l_1, an upper limit l_4, a lower support limit l_2, and an upper support limit l_3, as per Eq (7):

$$TS(x)[l_1, l_2, l_3, l_4] = \begin{cases} 0 & x \leq l_1 \\ (x - l_1)/(l_2 - l_1) & l_1 < x < l_2 \\ 1 & l_2 \leq x \leq l_3 \\ (l_4 - x)/(l_4 - l_3) & l_3 < x < l_4 \\ 0 & l_4 \leq x \end{cases} \tag{7}$$

Each FTW T_k is described by a trapezoidal function based on the time interval from a previous time t_i to the current time t^*: $T_k(\Delta t_i^*)[l_1, l_2, l_3, l_4]$. In order to generate FTWs in a simple manner, we propose to define them from a set of incrementally ordered evaluation times $L = \{L_1, \ldots, L_{|L|}\}, L_{i-1} < L_i$, where the limits of the trapezoidal functions are calculated according to the index of the temporal window T_k.

$$T_k = T_k(\Delta t_i^*)[L_k, L_{k-1}, L_{k-2}, L_{k-3}] \tag{8}$$

Once the FTWs $\{T_0, \ldots, T_{|T|}\}$ have been defined by incrementally ordered times, we apply the fuzzification to each time-slot in the timeline $t^+ \in T$ and sensor activation $S(t^+, s)$ as per Eq. (5). Subsequently it generates a *feature vector* of components $T_k(s, t^+)$ for each time-slot in the timeline t^+, the size of which is equal to the number of FTWs multiplied by the number of sensors $|T| \times |S|$.

As the semantic of windows provides a description from short to long temporal components, we can define an order sequence of aggregated degrees from the sensor activations within fuzzy temporal windows for each given time-slot in the timeline t^+ and the sensor s:

$$T(s, t^+) = \{T_0(s, t^+) \rightarrow \ldots \rightarrow T_k(s, t^+) \rightarrow \ldots \rightarrow T_{|T|}(s, t^+)\} \tag{9}$$

Table 3: An example of fuzzy temporal windows described by incrementally ordered evaluation times $L = \{720, 540, 360, 180, 60, 30, 15, 5, 3, 2, 1\}$ mins.

FTW	L_k	L_{k-1}	L_{k-2}	L_{k-3}
T_1	720	540	360	180
T_2	540	360	180	60
T_3	360	180	60	30
T_4	180	60	30	15
T_5	60	30	15	5
T_6	30	15	5	3
T_7	15	5	3	2
T_8	5	3	2	1
T_9	3	2	1	0
T_{10}	2	1	0	0

In Table 3, we show an example of FTWs described by incrementally ordered evaluation times L={720,540,360,180,60,30,15,5,3,2,1} min.

3.3.1. *Ensemble of classifiers for activities*

In this Section, an ensemble of LSTMs for the purpose of AR is proposed. The main concept is learning each activity in an isolated manner using an LSTM activity-based classifier.

With this approach, each LSTM activity-based classifier is focused on learning a given activity A_i by means of a balanced training dataset. In this way, from the same training dataset, several adapted-activity datasets can be built, namely one for each LSTM activity-based classifier. Each classifier is trained to recognise a particular activity and has a binary output to represent: A_i) when the target activity is presented, and, $\overline{A_i}$) when the target activity is not presented. This last output represents other activities and the idle activity. Moreover, the weight of each activity in the dataset is defined by similarity of the target activity to another and is discussed in Sections 3.3.2 and 3.3.3.

The inputs of the LSTM activity-based classifier for a given activity A_i are composed of a given time-slot t^+ and the ensuing related information:

The target activity $O(t^+)$ is defined by:

$$O(t^*) = \begin{cases} 1 & S(t^+, A_i) == 1 \\ 0 & S(t^+, A_i) \neq 1 \end{cases} \tag{10}$$

The *feature vector* is formed by the sequence of aggregated degrees of activation $T_k(s, t^+)$

within the FTWs T_k for each sensor s for a given time-slot t^+, the size of which is equal to the number of FTW multiplied by the number of sensors $|T| \times |S|$.

Once the learning process is complete, in the testing phase, the activation of the target activity A_i, which has been learned by its LSTM activity-based classifier, is presented when the prediction for being target activity p_{A_i} overcomes the prediction of not being the target activity $p_{\overline{A_i}}$. In order to provide a normalized degree of activation between $[0, 1]$, the softmax function as a normalized exponential function (Bishop, 2006), has been applied to output prediction as expressed by the following equation:

$$\mu_{p_{A_i}} = \frac{e^{p_{A_i}}}{e^{p_{A_i}} + e^{p_{\overline{A_i}}}} \tag{11}$$

In case of conflict between classifiers within the ensemble, when several activities are detected at the same time, the maximal value of prediction from the LSTM activity-based classifiers is selected $A(t^+)^*$ as the activity carried out by the inhabitant in the given time-slot t^+.

$$A(t^+)^* = \begin{cases} A_0 & \mu_{p_{A_i}} = 0, \forall A_i \in |A| \\ A_i, \mu_{p_{A_i}} > \mu_{p_{A_j}} \forall A_i, A_j \in |A|, A_j \neq A_i & \text{otherwise} \end{cases} \tag{12}$$

We note that this step is included when the dataset is formed by activities without overlapping, however, avoiding this step could provide multiple activations in the context of interleaved activities.

In Figure 4, a scheme for the architecture of the ensemble of LSTM activity-based classifiers is presented.

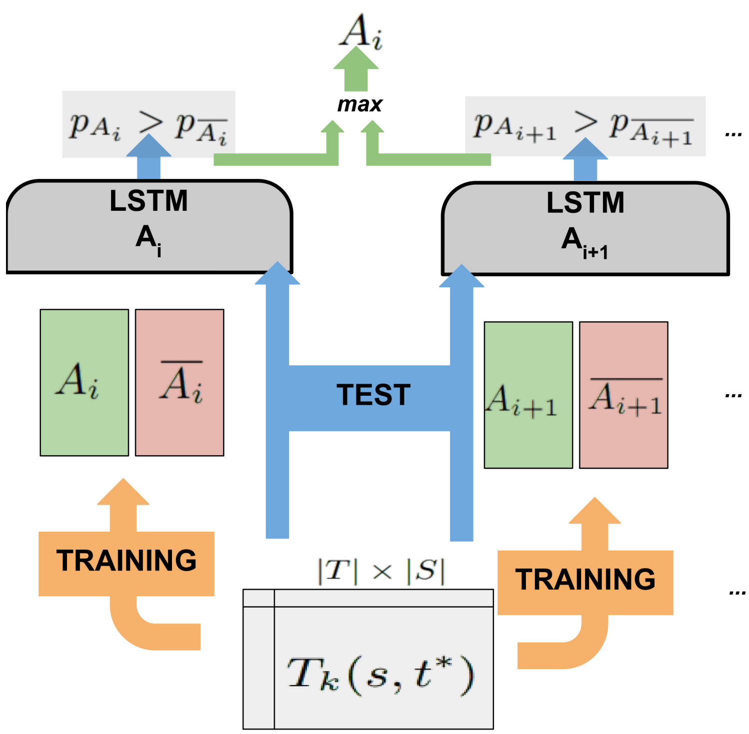

Figure 4: Architecture of the ensemble of LSTM activity-based classifiers

In the following sections, we describe how to build an *ad hoc* balanced training dataset for each activity-based classifier based on the similarity relation to other activities.

3.3.2. Computing similarity relation between activities based on sensor activation

Based on a given activity A_i and another activity A_j, we define a similarity relation R_a as a function $R_a : A_i \times A_j \rightarrow [0, 1]$ which determines the degree of similarity between both activities. Next, we describe an approach to compute the similarity based on the frequency of common sensors.

Firstly, from the segmented timeline, we calculate a similarity relation $R_s : A_i \times S_j \rightarrow [0, 1]$ between activities and the sensor using the relative frequency of the sensor activation within each activity:

$$R_s(A_i, S_j) = \frac{|S_j \cap A_i|}{\sum_{S_k}^{S} |S_j \cap A_i|}$$
$$|S_j \cap A_i| = \sum_{t^+}^{T} \begin{cases} 1 & S(t^+, A_i) = S(t^+, S_j) \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

where $|S_j \cap A_i|$ represents the number of time-slots activated when the sensor S_j is activated together with the activity A_i. This measure is also called *Mutual Information* Krishnan & Cook (2014).

Secondly, we evaluate the similarity relation between activities R_a aggregating the similarity relation from their sensor activations:

$$R_a(A_i, A_j) = \sum_{S_k}^{S} R_a(A_i, S_k) \times R_a(A_j, S_k), A_i \neq A_j \tag{14}$$

We note that we can normalize the degree of similarity for a given activity A_i, $\widetilde{R_a}(A_i, A_j) = \frac{R_a(A_i,A_j)}{\sum_{A_k}^{A} R_a(A_i,A_k)}$. In the next Section, we detail how to balance the training dataset based on the degree of similarity in the ensemble of activity classifiers.

3.3.3. Balancing training in the ensemble of activity classifiers

In this Section, we describe how to balance the training dataset for each activity classifier in order to: i) solve the imbalance problem within datasets, ii) to obtain a more representative training dataset for conflicting activities, obtaining a higher similarity relation.

In this way, we based the balancing of the training data on the similarity relation between activities. Specifically, we propose to build a *balanced-activity training dataset*, which contains a ratio of samples (weight) for each activity A_i:

- w_{A_i}, which represents the ratio in the balanced-activity training dataset corresponding to the activity to be learned.
- w_{A_0}, which represents the ratio of samples in the balanced-activity training dataset, corresponding to any activity (Idle).
- $w_{\overline{A_i}}$, which configures the ratio of other activities in the balanced-activity training dataset $w_{A_i} + w_{A_0} + w_{\overline{A_i}} = 1$. The ratio for all of the other activities is calculated by weighting the normalized degree of similarity with the ratio of the other activities:

$$w_{A_j} = w_{\overline{A_i}} \times \widetilde{R_a}(A_i, A_j) \tag{15}$$

  We note that $w_{\overline{A_i}}$ is defined as a fixed ratio for all other activities, which is weighted for each other activity based on the similarity guaranteeing that finally $w_{A_0} + w_{A_1} + \ldots + w_{A_n} = 1$.

  A minimal ratio w_{min} of time-slots per activity, in case a close-to-zero degree of similarity is obtained, is recommended in order to guarantee a minimal representation.

In order to select the time-slots, a random process, which is weighted by the previously defined ratios, is established. This process selects a time-slot randomly, rejecting or accepting them based on the current ratio of the activities which are activated within it. In Algorithm 1, we detail the pseudo-code of this process.

```
Data: {w_{A_0}, w_{A_1}, ...}, N
Result: ts = {... t_k, ...}
c'_{A_{1...|A|}} = 0;
count = 0;
ts = ∅;
while count < N do
    t_k = randomIndex(t_0, ..., t_{|T|});
    forall A_i ∈ A_i(t_k) = 1 do
        w'_{A_i} = 0;
        if count > 0 then
            w'_{A_i} = c'_{A_i}/count;
        end
        if w'_{A_i} < w_{A_i} then
            ts = ts ∪ t_k;
            c'_{A_i} = c'_{A_i} + 1;
            count + +;
            if count >= N then
                break;
            end
        end
    end
end
```

Algorithm 1: Algorithm for obtaining random time-slots from a balanced ratio of activities $\{w_{A_0}, w_{A_1}, \dots\}$

For a time-slot t_k, which is obtained from the function *randomIndex* selecting a random time-slot in the timeline, we evaluate if the computed ratio $w'_{A_i} = c'_{A_i}/count$ of the activities activated A_i in the time t_k does not overcome the threshold ratio w_{A_i}, in which case we accept the time-slot t_k. At the end, when the target number of samples N is collected (and even in each iteration) the method guarantees that the ratio of selected time-slots per activity remains under the defined threshold ratio of the activities.

4. Evaluation

In this Section, the experiments performed according to the proposed methodology are evaluated using two popular datasets: Ordoñez (Ordónez et al., 2013) and CASAS (Cook & Schmitter-Edgecombe, 2009; Singla, Cook & Schmitter-Edgecombe, 2009), where the binary sensor activation is related to the daily activities of an inhabitant labelled by an external observer.

In both datasets, the methodology proposed in this work defines the following parameters:

- Number of FTWs=$|T| = 10$.
- Incrementally ordered evaluation times $L = \{720, 540, 360, 180, 60, 30, 15, 5, 3, 2, 1\} \cdot \Delta t$. L is defined by a human expert to relate to short, mid and long intervals of time from minutes to hours.
- For balancing the training dataset for each activity:
  - Number of training samples = 10,000.
  - Weight of samples corresponding to the target activity $w_{A_i} = 0.4$.
  - Weight of samples corresponding to the idle activity $w_{A_0} = 0.1$.
  - Weight of samples corresponding to the non-target activity $w_{\overline{A_i}} = 0.6$.
  - Minimal ratio of similarity per activity $w_{min} = 0.05$.
- For each LSTM activity-based classifier:
  - Learning rate = 0.0001, to work well as a standard parameterization (Salimans & Kingma, 2016; Wu, Zhang, Zhang, Bengio & Salakhutdinov, 2016).
  - Number of neurons = 64, as a minimal reference value in learning patterns in RNN (De Pietro, Gallo, Howlett & Jain, 2018).
  - Number of layers = 3, due to a great number of layers increasing learning time exponentially without significance in accuracy (Collins, Sohl-Dickstein & Sussillo, 2017).

– Training epochs = 40 and batch size = 1000 to complete almost 4 iterations over the training dataset (training samples = 10000).

The following three popular metrics are used to evaluate the datasets:

- Accuracy (acc), which represents the correctly classified percentage, TP being true positives, TN, true negatives, FP, false positives and, finally, FN, fase negatives: $acc = \frac{TP+TN}{TP+TN+FP+FN}$.

  This metric has been used in other related works (Singh et al., 2017), however, in learning situations using imbalanced datasets, the overall classification accuracy is not considered as an appropriate measure of performance.

- F1-score (*F1-sc*), which provides an insight into the balance between precision, which is $precision = \frac{TP}{TP+FP}$, and recall, which is $recall = \frac{TP}{TP+FN}$. Although this metric is well-known in AR (Van Kasteren et al., 2010), we note a key issue from this metric on time interval analysis: the FPs of an activity far from any time interval activation are equally computed to false positives closer to the end of activities, which are common in the end of activities (refer to Figure 5). For taking into account a time interval evaluation, we propose the following additional metric.

- F1-interval-intersection (*F1-ii*), which evaluates the time intervals of each activity based on: i) the precision of predicted time intervals which intersect with a ground truth time interval, ii) the recall of the ground truth time intervals which intersect with a predicted time interval.

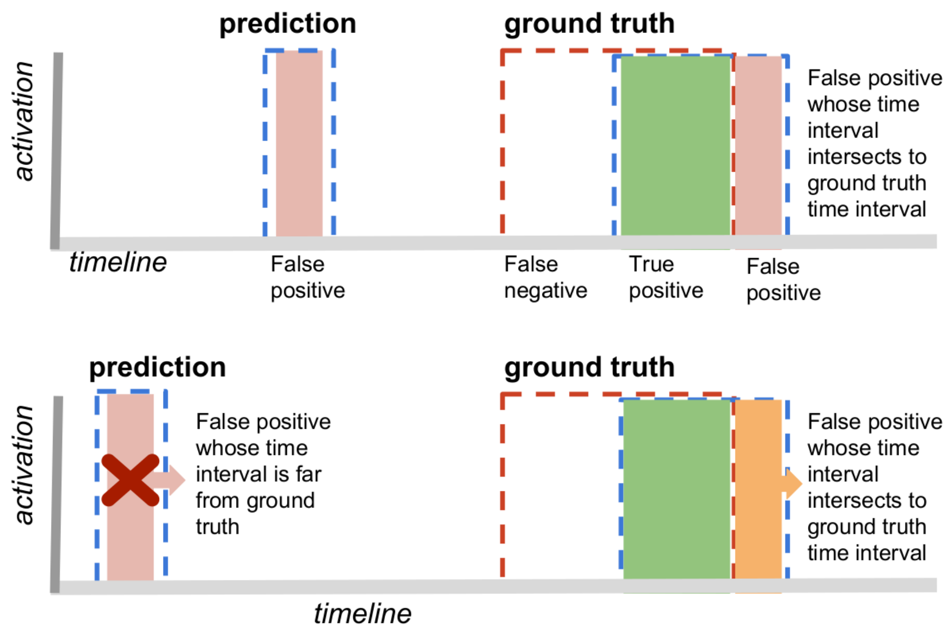

Figure 5: Evaluation and metrics of predicted and ground truth activity time intervals in AR.

4.1. Ordoñez Dataset

In this dataset, two experiments were carried out in different rooms (A and B). In room A, 12 binary sensors describe 14 days where 9 daily activities were carried out over a period of 19,932 minutes. In room B, 12 binary sensors describe 22 days where 10 daily activities are carried out over a period of 30,495 minutes.

We have initially segmented the timeline in time-slots using the window size $\Delta t = 60s$, based on the standard reference from (van Kasteren et al., 2011; Ordóñez & Roggen, 2016; Singh et al., 2017).

For evaluation purposes, we have developed a leave-one-day cross-validation, where the test is composed by a single day and training is composed by other days. Then, the process is repeated, selecting different test days and finally iterating over all days. (Van Kasteren et al., 2008). To include a complete day cycle without splitting a short activity, we have taken into account the days between 4:00pm and the following 24 hours. After obtaining the results of classification with leave-one-day cross-validation, other authors have provided an average of the metrics between days (Ordóñez & Roggen, 2016; Singh et al., 2017). Nevertheless, initial and end days are usually shorter and many days do not include the development of all activities, which can undermine the total performance. To solve this issue, we have merged all time-slots from leave-one-day tests configuring a timeline test per activity which can be analyzed by the metrics.

4.2. CASAS Dataset

This dataset includes two experiments of activities which were performed individually and sequentially by several inhabitants. Experiment A (Singla et al., 2009) contains 8 activities carried out by 21 inhabitants with a total of 10.334 binary sensor records, and experiment B (Cook & Schmitter-Edgecombe, 2009) contains 5 activities carried out by 51 inhabitants with a total of 6.425 binary sensor records.

We have evaluated three window sizes for segmenting the timeline in time-slots $\Delta t = \{5s, 20s, 60s\}$. We note in the Ordoñez dataset that the optimal window size for time-slots was fixed by the evaluation of previous works (van Kasteren et al., 2011; Ordóñez & Roggen, 2016; Singh et al., 2017).

For evaluation purposes we have developed a leave-one-inhabitant cross-validation, where for each inhabitant, the test is composed of activities developed by the given inhabitant and training is composed of the activities developed by other inhabitants. After obtaining the classification results from the leave-one-inhabitant cross-validation, we have merged all time-slots from the tests configuring a timeline test per activity which can be analyzed by the metrics.

4.3. Results

In this Section, we compare the results of the proposed methodology with the two datasets and results from previous works. In (Ordónez et al., 2013), the representation of sensor by means of raw and last activation was demonstrated to provide encouraging results in real-time AR by means of non-sequence classifiers, highlighting Support Vector Machines (SVM) and Decision Trees (C4.5), which have been compared against the methodology proposed in this work.

As previously detailed, all time-slots from leave-one cross-tests have been merged configuring a timeline test per activity which can be analyzed with the metrics. For each metric (accuracy, F1-score and F1-interval-intersection), we have analyzed the average of the metric per activity in the timeline test. To avoid the possible effect of imprecision when segmenting the dataset into time-slots, a confidence interval of one time-slot is allowed in computing the F1-score and F1-interval-intersection.

The results from the experiments within the Ordoñez dataset (Room A and Room B) are described in Table 4 and 5.

In the CASAS dataset, we have evaluated three time intervals of $\Delta t = \{5s, 20s, 60s\}$ in length. In Table 6, we present the average of the metrics for each approach and time interval.

Table 4: Metrics expressed by percentage for real-time AR in Ordoñez Room A

	FTW+LSTM			Raw+Last+SVM			Raw+Last+C4.5		
	Acc	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**
Leaving	99.53	90.32	97.26	99.75	96.55	98.56	99.75	96.55	98.56
Toileting	98.73	72.96	51.74	98.89	86.74	38.09	98.89	80.52	35.19
Showering	99.92	100	94.02	99.96	100	96.43	99.96	100	96.43
Sleeping	99.99	100	99.99	99.98	100	99.98	99.99	100	99.99
Breakfast	99.78	100	85.71	99.72	77.0	81.76	99.74	80	82.71
Lunch	99.55	90.00	86.98	99.53	72	82.26	99.54	72	86.49
Snack	99.97	80.0	80	99.92	40.0	40.0	99.92	0	0
Spare Time	98.16	98.99	97.91	98.73	98.99	98.56	98.71	97.96	98.53
Grooming	99.57	91.36	76.92	99.58	79.54	75.22	99.57	79.57	75.21
Total	99.47	**91.51**	**85.61**	**99.56**	83.51	79.43	99.56	78.51	74.79

Table 5: Metrics expressed by percentage for real-time AR in Ordoñez Room B

	FTW+LSTM			Raw+Last+SVM			Raw+Last+C4.5		
	Acc	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**
Leaving	98.09	90.47	94.69	99.54	98.67	98.68	99.53	98.67	98.66
Dinner	99.53	47.61	28	99.60	0	0	99.58	21.05	11.11
Toileting	99.83	89.11	85.39	99.83	94.18	84.78	99.85	97.17	88.0
Showering	99.96	90.91	92.31	99.98	95.62	95	99.98	95.65	95.54
Sleeping	98.65	92.15	98.08	98.87	96.35	98.34	98.89	98.04	98.41
Breakfast	99.30	81.18	69.52	98.49	33.33	29.36	98.56	52.53	41.64
Lunch	98.04	45.55	40.87	98.63	49.48	13.28	98.62	45.28	13.20
Snack	97.32	46.86	30.85	98.18	42.11	17.26	98.32	42.70	13.54
Spare Time	94.84	84.17	91.28	94.94	65.86	91.15	94.83	66.28	90.97
Grooming	99.54	93.23	85.77	99.71	98.41	90.31	99.73	97.33	90.79
Total	98.51	**76.12**	**71.68**	98.77	67.40	61.99	**98.79**	71.47	64.19

Moreover, in Tables 7 and 8, we detail the activity performance for each approach in its best time interval for Experiment A and B, respectively.

It is noteworthy that the data and code from the experiments and results described are

Table 6: Total metrics expressed by percentage for time intervals in the CASAS dataset $\Delta t = \{5s, 20s, 60s\}$

		FTW+LSTM			Raw+Last+SVM			Raw+Last+C4.5		
Exp	Δt	**Acc**	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**
A	5s	**97.97**	**95.79**	**90.59**	95.35	52.03	77.80	95.66	51.00	79.72
A	20s	**97.68**	**97.56**	**89.67**	95.86	68.83	82.91	95.82	66.84	82.39
A	60s	**97.27**	**94.69**	**88.19**	94.96	88.98	81.21	94.35	82.30	78.96
B	5s	**95.47**	**92.67**	**84.98**	90.25	46.43	60.27	90.45	47.47	63.71
B	20s	**95.73**	**97.48**	**88.62**	91.04	65.20	74.47	90.30	63.02	71.12
B	60s	**95.98**	**96.89**	**90.32**	90.05	80.68	72.57	90.20	76.91	71.28

Table 7: Metrics expressed by percentage for real-time AR in CASAS Experiment A

	FTW+LSTM			Raw+Last+SVM			Raw+Last+C4.5		
	$\Delta t = 20s$			$\Delta t = 60s$			$\Delta t = 60s$		
	Acc	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**
Medication	99.61	97.44	97.96	96.41	77.55	83.91	96.54	76.00	84.39
Watch	95.79	97.56	90.63	92.32	80.00	84.46	91.29	65.71	82.83
Water plants	98.73	95.24	88.70	95.52	90.91	72.87	93.21	70.83	60.15
Phone	97.16	95.00	81.53	95.01	91.30	76.36	95.26	93.02	76.73
Prepare card	96.18	100.00	87.30	95.52	89.36	85.83	96.16	87.50	87.70
Cook	97.75	97.67	93.90	93.73	93.33	84.44	93.47	89.36	84.01
Clean	97.55	97.56	90.57	93.98	89.36	81.12	91.93	78.43	74.90
Choose outfit	98.63	100.00	86.79	97.18	100.00	80.70	96.93	97.56	80.95
Total	**97.68**	**97.56**	**89.67**	94.96	88.98	81.21	94.35	82.30	78.96

shared under Creative Commons Attribution 3.0 in the following repository [1].

4.4. Discussion

From the results previously described, we note the general improvement in terms of the F1-score and F1-intersection of time intervals in real-time AR.

When considering the Ordoñez dataset, we particularly note the improvement in terms of F1-ii, which indicates that the temporal intervals predicted by LSTM are closer in terms of temporal

[1]http://serezade.ujaen.es:8054/lstm-ftw/

Table 8: Metrics for real-time AR in CASAS Experiment B

	FTW+LSTM			Raw+Last+SVM			Raw+Last+C4.5		
	$\Delta t = 20s$			$\Delta t = 60s$			$\Delta t = 60s$		
	Acc	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**	**Acc**	**F1-ii**	**F1-sc**
Phone	99.44	100	97.98	95.48	100	85.21	95.48	100	85.21
Wash	99.02	100	92.47	91.68	57.14	43.90	91.32	39.02	36.84
Cook	93.96	100	92.20	88.25	84.74	85.06	88.79	85.71	85.71
Eat	91.43	91.39	69.95	89.15	90.57	71.70	89.33	86.79	71.22
Clean	94.80	96.00	90.49	85.71	70.97	76.97	86.08	73.02	77.42
Total	**95.73**	**97.48**	**88.62**	90.05	80.68	72.57	90.20	76.91	71.29

distance to the ground truth. On the accuracy metric, previous works present a slightly better performance due to LSTM discriminates sensor activations by more than one temporal window, where F1-ii and F1-sc are higher. Moreover, LSTM analyzes more temporal features waiting for the permanence of sensor activation within several temporal windows to overcome the certainty of an activity being developed. We note that accuracy has been shown not to be a representative metric in AR (Ordónez et al., 2013).

Moreover, in Ordoñez Room A, we highlight the high performance of conflicting activities in the kitchen: *Lunch, Breakfast and Snack*. This is due to the integration of long-mid temporal information and the *ad hoc* balanced learning included for each classifier per activity. This impact is more prominent in the case of *Dinner* in dataset Ordoñez Room B, which represents a scarce activity which is just presented for a few days, lasting a short time and with several conflicting activities: *Lunch, Breakfast and Snack*. This case is so difficult to analyse that *raw+last+svm* is not able to detect the dinner activity; however, *FTW+LSTM* has collected the proper information from long-mid term to generate a notable increase of classification which duplicates its performance.

In the CASAS dataset, the performance of *FTW+LSTM* notably overcomes the performance of the feature representation of raw and last activation for real-time AR. Here, the accuracy, F1-ii and F1-sc are increased due to the sequence learning and temporal aggregation of sensor activation developed by FTWs. Moreover, a relevant strength of our approach is the robust performance of the variation of the window size in time-slots segmenting the timeline. As in $\Delta t = \{5s, 20s, 30s\}$ the methodology presents an encouraging recognition of activities, highlighting F1-ii which represents the prediction within the temporal intervals of ground truth. So, unlike the previous works, our methodology is able to provide a response in real-time with a finer granularity of time.

Table 9 presents a comparison between relevant windowing methodologies, some of which are lacking real-time capabilities. We note the performance measures of some methodologies are not directly comparable; for example, in (Shahi et al., 2017) the segmentation is based on events instead of time-slots, and the testing is only developed in 20% of the data. In (Espinilla et al., 2018) the authors propose two learning layers under a windowing approach i) to evaluate the ending point of the activities without real-time AR, and ii) after a selection of the most suitable window size. In these cases, although the performance measures of these methodologies are not directly comparable, it is key that the time interval recognition of activities (F1-ii) in the timeline from this work obtains a higher performance than accuracy in AR under the windowing approach; notwithstanding that evaluating the ending point of the activities provides the most advantageous point of time in AR without offering real-time capabilities.

Table 9: Metrics, results and window approaches in evaluated datasets

Dataset	**Reference works**	**Metric / Value**	**Learning**	**Features**	**Window**
CASAS B	(Espinilla et al., 2016)	Acc (96.67%)	Prototype+KNN	raw	window from offline human labelled observation (explicit segmentation)
CASAS B	**FTW+LSTM**	F1-ii (97.56%)	LSTM	FTW	$\Delta t = 20s$ (real-time based on time-slots)
CASAS B	(Shahi et al., 2017)	F1-sc(95.3%) Acc (87.5%)	NB	mutual information + time interval	dynamic window (real-time based on events)
CASAS B	**FTW+LSTM**	F1-sc(88.6%) Acc (95.7%)	LSTM	FTW+time-slots (20s)	FTW $\Delta t = 20s$ (real-time based on time-slots)
Ordoñez A	(Espinilla et al., 2018)	Acc (89.1%)	DT (C4.5)	Dynamic window + 3 subwindows	Analysis at ending point of activity (partial explicit segmentation)
Ordoñez A	**FTW+LSTM**	F1-ii (91.51%)	LSTM	FTW	$\Delta t = 60s$ (real-time based on time-slots)
Ordoñez B	(Ordónez et al., 2013)	F1-sc (64.19%)	DT (C4.5)	raw+last activation	$\Delta t = 60s$ (real-time based on time-slots)
Ordoñez B	**FTW+LSTM**	F1-sc (79.43%)	LSTM	FTW	$\Delta t = 60s$ (real-time based on time-slots)

The achievements of this work, which aggregates long-term information from binary sensors by means of incremental FTW using the LSTM sequence classifier, have reduced the complexity of i) fixing the window sizes for each activity, ii) selecting an optimal time interval to segment the information of binary sensors and iii) defining human-defined features.

5. Conclusions and future works

In this work, we have presented a methodology to aggregate binary sensor activations using Fuzzy Temporal Windows. This approach has been evaluated as a suitable representation using a fuzzy temporal aggregation as described in a previous work (Medina et al., 2017c). In summary, defining multiple FTWs of incremental size from long-term to short-term has provided an adequate semantic to define a sequence of temporal features, which has been suitable for learning using the LSTM sequence classifier.

In this work, LSTM has been demonstrated as a powerful classifier to understand the raw activation between sensors and without requiring additional representations in the feature vector based on external knowledge, such as the last sensor activation. Moreover, a metric to evaluate the temporal distance of predicted time intervals to ground truth has also been introduced.

In addition, the use of the ensemble of LSTM activity-based classifiers could provide a straightforward adaptation to complex contexts, such as interleaved activities (Singla et al., 2009).

Furthermore, as we have discussed, the mosaic of approaches in Activity Recognition is wide. Given the convergence of different technologies in Activity Recognition, in future work we will include wearable devices to detect user interaction with daily objects by means of proximity sensors (Medina et al., 2017a). This approach could include the advantage of privacy in addition to facing multi-occupancy in real-time AR. In order to introduce the use of FTW in non-binary sensors, such as wearable devices, we note that defining aggregated features represents a more complex problem than fuzzy temporal aggregation for binary sensors. In this way, several feature selection methods would be necessary to extract long and mid temporal patterns within fuzzy time intervals defined by the temporal windows. These open issues will be analyzed as the next step of future works.

Acknowledgment

This research has received funding under the REMIND project Marie Sklodowska-Curie EU Framework for Research and Innovation Horizon 2020, under Grant Agreement No. 734355. Invest Northern Ireland is acknowledged for partially supporting this project under the Competence Centre Programme Grant RD0513853 - Connected Health Innovation Center. This contribution has been supported as well by the PI-0203-2016 project from the Council of Health for the Andalusian Health Service (Spain) together with the Spanish governments TIN2015-66524-P research project.

References

Alcalá, J. M., Ureña, J., Hernández, Á., & Gualda, D. (2017). Sustainable homecare monitoring system by sensing electricity data. *IEEE Sensors Journal*, *17*, 7741–7749.

Banos, O., Galvez, J.-M., Damas, M., Pomares, H., & Rojas, I. (2014). Window size impact in human activity recognition. *Sensors*, *14*, 6474–6499.

Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM Sigkdd Explorations Newsletter*, *6*, 20–29.

Belley, C., Gaboury, S., Bouchard, B., & Bouzouane, A. (2013). Activity recognition in smart homes based on electrical devices identification. In *Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments* (p. 7). ACM.

Bishop, C. M. (2006). Pattern recognition and machine learning (information science and statistics) springer-verlag new york. *Inc. Secaucus, NJ, USA*, .

Carnevali, L., Nugent, C., Patara, F., & Vicario, E. (2015). A continuous-time model-based approach to activity recognition for ambient assisted living. In *International Conference on Quantitative Evaluation of Systems* (pp. 38–53). Springer.

Catal, C., Tufekci, S., Pirmit, E., & Kocabag, G. (2015). On the use of ensemble of classifiers for accelerometer-based activity recognition. *Applied Soft Computing*, *37*, 1018–1022.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321–357.

Chen, L., Hoey, J., Nugent, C. D., Cook, D. J., & Yu, Z. (2012). Sensor-based activity recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *42*, 790–808.

Collins, J., Sohl-Dickstein, J., & Sussillo, D. (2017). Capacity and trainability in recurrent neural networks. *stat*, *1050*, 28.

Cook, D., Schmitter-Edgecombe, M., Crandall, A., Sanders, C., & Thomas, B. (2009). Collecting and disseminating smart home sensor data in the casas project. In *Proceedings of the CHI Workshop on Developing Shared Home Behavior Datasets to Advance HCI and Ubiquitous Computing Research* (pp. 1–7).

Cook, D. J., & Schmitter-Edgecombe, M. (2009). Assessing the quality of activities in a smart environment. *Methods of information in medicine*, *48*, 480.

De Pietro, G., Gallo, L., Howlett, R. J., & Jain, L. C. (2018). *Intelligent Interactive Multimedia Systems and Services 2017*. Springer.

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine learning*, *40*, 139–157.

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2625–2634).

Edwards, T. J., Coward, J., Hamer, P., Twitchen, K. J., & Hobson, P. W. (2000). Method and apparatus for forecasting future values of a time series. US Patent 6,125,105.

Espinilla, M., Liu, J., & Chamizo, J. (2017a). Special issue on recent advancements in ubiquitous computing. *Journal of Ambient Intelligence and Humanized Computing*, *8*, 467–468.

Espinilla, M., Medina, J., Calzada, A., Liu, J., Martinez, L., & Nugent, C. (2017b). Optimizing the configuration of an heterogeneous architecture of sensors for activity recognition, using the extended belief rule-based inference methodology. *Microprocessors and Microsystems*, *52*, 381–390.

Espinilla, M., Medina, J., Hallberg, J., & Nugent, C. (2018). A new approach based on temporal sub-windows for online sensor-based activity recognition. *Journal of Ambient Intelligence & Humanized Computing*, *In Press*.

Espinilla, M., & Nugent, C. (2017). Computational intelligence for smart environments. *International Journal of Computational Intelligence Systems*, *10*, 1875–6883.

Espinilla, M., Rivera, A., Pérez-Godoy, M. D., Medina, J., Martinez, L., & Nugent, C. (2016). Recognition of activities in resource constrained environments; reducing the computational complexity. In *Ubiquitous Computing and Ambient Intelligence* (pp. 64–74). Springer.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2012). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *42*, 463–484.

Gu, T., Wang, L., Wu, Z., Tao, X., & Lu, J. (2011). A pattern mining approach to sensor-based human activity recognition. *IEEE Transactions on Knowledge and Data Engineering*, *23*, 1359–1372.

Guo, X., Yin, Y., Dong, C., Yang, G., & Zhou, G. (2008). On the class imbalance problem. In *Natural Computation, 2008. ICNC'08. Fourth International Conference on* (pp. 192–201). IEEE volume 4.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*, 1735–1780.

Huynh, T., Blanke, U., & Schiele, B. (2007). Scalable recognition of daily activities with wearable sensors. In *LoCA* (pp. 50–67). volume 7.

Japkowicz, N., & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent data analysis*, *6*, 429–449.

van Kasteren, T. L. M. et al. (2011). *Activity recognition for health monitoring elderly using temporal probabilistic models*. Ph.D. thesis Universiteit van Amsterdam.

Koutitas, G. C., & Tassiulas, L. (2016). Low cost disaggregation of smart meter sensor data. *IEEE Sensors Journal*, *16*, 1665–1673.

Krishnan, N. C., & Cook, D. J. (2014). Activity recognition on streaming sensor data. *Pervasive and mobile computing*, *10*, 138–154.

Krüger, F., Nyolt, M., Yordanova, K., Hein, A., & Kirste, T. (2014). Computational state space models for activity and intention recognition. a feasibility study. *PloS one*, *9*, e109381.

Lester, J., Choudhury, T., Kern, N., Borriello, G., & Hannaford, B. (2005). A hybrid discriminative/generative approach for modeling human activities. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*.

Logan, B., Healey, J., Philipose, M., Tapia, E. M., & Intille, S. (2007). A long-term evaluation of sensing modalities for activity recognition. In *International conference on Ubiquitous computing* (pp. 483–500). Springer.

Martin, J. (1965). *Programming real-time computer systems*. Technical Report.

Mazurowski, M. A., Habas, P. A., Zurada, J. M., Lo, J. Y., Baker, J. A., & Tourassi, G. D. (2008). Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural networks*, *21*, 427–436.

Medina, J., Espinilla, M., & Nugent, C. (2016). Real-time fuzzy linguistic analysis of anomalies from medical monitoring devices on data streams. In *Proceedings of the 10th EAI International Conference on Pervasive Computing Technologies for Healthcare* (pp. 300–303). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Medina, J., Espinilla, M., Zafra, D., Martínez, L., & Nugent, C. (2017a). Fuzzy fog computing: A linguistic approach for knowledge inference in wearable devices. In *International Conference on Ubiquitous Computing and Ambient Intelligence* (pp. 473–485). Springer.

Medina, J., Fernandez-Olmo, M., Peláez, M., & Espinilla, M. (2017b). Real-time monitoring in home-based cardiac rehabilitation using wrist-worn heart rate devices. *Sensors*, *17*. doi:10.3390/s17122892.

Medina, J., Martinez, L., & Espinilla, M. (2017c). Subscribing to fuzzy temporal aggregation of heterogeneous sensor streams in real-time distributed environments. *International Journal of Communication Systems*, *30*.

Möller-Acuña, P., Ahumada-García, R., & Reyes-Suárez, J. A. (2017). Machine learning for prediction of frost episodes in the maule region of chile. In *International Conference on Ubiquitous Computing and Ambient Intelligence* (pp. 715–720). Springer.

Ordóñez, F. J., & Roggen, D. (2016). Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors*, *16*, 115.

Ordónez, F. J., de Toledo, P., & Sanchis, A. (2013). Activity recognition using hybrid generative/discriminative models on home environments using binary sensors. *Sensors*, *13*, 5460–5477.

Ortiz, J. L. R. (2015). *Smartphone-based human activity recognition*. Springer.

Patterson, T., Khan, N., McClean, S., Nugent, C., Zhang, S., Cleland, I., & Ni, Q. (2017). Sensor-based change detection for timely solicitation of user engagement. *IEEE Transactions on Mobile Computing*, *16*, 2889–2900.

Rege, A., Mehra, S., Vann, A., & Luo, Z. (2017). Vision-based approach to senior healthcare: Depth-based activity recognition with convolutional neural networks. *Semantic Scholar*, .

Reyes-Ortiz, J.-L., Anguita, D., Ghio, A., & Parra, X. (2012). Human activity recognition using smartphones data set. *UCI Machine Learning Repository; University of California, Irvine, School of Information and Computer Sciences: Irvine, CA, USA*, .

Robertson, N., & Reid, I. (2006). A general method for human activity recognition in video. *Computer Vision and Image Understanding*, *104*, 232–248.

Roggen, D., Calatroni, A., Rossi, M., Holleczek, T., Förster, K., Tröster, G., Lukowicz, P., Bannach, D., Pirkl, G., Ferscha, A. et al. (2010). Collecting complex activity datasets in highly rich networked sensor environments. In *Networked Sensing Systems (INSS), 2010 Seventh International Conference on* (pp. 233–240). IEEE.

Rossetti, G., Milli, L., Giannotti, F., & Pedreschi, D. (2017). Forecasting success via early adoptions analysis: A data-driven study. *PloS one*, *12*, e0189096.

Salimans, T., & Kingma, D. P. (2016). Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems* (pp. 901–909).

San Martín, L. Á., Peláez, V. M., González, R., Campos, A., & Lobato, V. (2010). Environmental user-preference learning for smart homes: An autonomous approach. *Journal of Ambient Intelligence and Smart Environments*, *2*, 327–342.

Sanchez, W., Martinez, A., & Gonzalez, M. (2017). Towards job stress recognition based on behavior and physiological features. In *International Conference on Ubiquitous Computing and Ambient Intelligence* (pp. 311–322). Springer.

Shahi, A., Woodford, B. J., & Lin, H. (2017). Dynamic real-time segmentation and recognition of activities using a multi-feature windowing approach. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 26–38). Springer.

Shewell, C., Medina-Quero, J., Espinilla, M., Nugent, C., Donnelly, M., & Wang, H. (2017). Comparison of fiducial marker detection and object interaction in activities of daily living utilising a wearable vision sensor. *International Journal of Communication Systems*, *30*.

Singh, D., Merdivan, E., Hanke, S., Kropf, J., Geist, M., & Holzinger, A. (2017). Convolutional and recurrent neural networks for activity recognition in smart environment. In *Towards Integrative Machine Learning and Knowledge Extraction* (pp. 194–205). Springer.

Singla, G., Cook, D. J., & Schmitter-Edgecombe, M. (2009). Tracking activities in complex settings using smart environment technologies. *International journal of biosciences, psychiatry, and technology (IJBSPT)*, *1*, 25.

Sixsmith, A., & Johnson, N. (2004). A smart sensor to detect the falls of the elderly. *IEEE Pervasive computing*, *3*, 42–47.

Srivastava, N., Mansimov, E., & Salakhudinov, R. (2015). Unsupervised learning of video representations using lstms. In *International conference on machine learning* (pp. 843–852).

Stikic, M., Huynh, T., Van Laerhoven, K., & Schiele, B. (2008). Adl recognition based on the combination of rfid and accelerometer sensing. In *Pervasive Computing Technologies for Healthcare, 2008. PervasiveHealth 2008. Second International Conference on* (pp. 258–263). IEEE.

Storf, H., Kleinberger, T., Becker, M., Schmitt, M., Bomarius, F., & Prueckner, S. (2009). An event-driven approach to activity recognition in ambient assisted living. In *European Conference on Ambient Intelligence* (pp. 123–132). Springer.

Tapia, E. M., Intille, S. S., & Larson, K. (2004). Activity recognition in the home using simple and ubiquitous sensors. In *International conference on pervasive computing* (pp. 158–175). Springer.

Van Kasteren, T., Englebienne, G., & Kröse, B. J. (2010). An activity monitoring system for elderly care using generative and discriminative models. *Personal and ubiquitous computing*, *14*, 489–498.

Van Kasteren, T., Noulas, A., Englebienne, G., & Kröse, B. (2008). Accurate activity recognition in a home setting. In *Proceedings of the 10th international conference on Ubiquitous computing* (pp. 1–9). ACM.

Wilson, D. H., & Atkeson, C. (2005). Simultaneous tracking and activity recognition (star) using many anonymous, binary sensors. In *International Conference on Pervasive Computing* (pp. 62–79). Springer.

Wu, Y., Zhang, S., Zhang, Y., Bengio, Y., & Salakhutdinov, R. R. (2016). On multiplicative integration with recurrent neural networks. In *Advances in Neural Information Processing Systems* (pp. 2856–2864).

Yala, N., Fergani, B., & Fleury, A. (2015). Feature extraction for human activity recognition on streaming data. In *Innovations in Intelligent SysTems and Applications (INISTA), 2015 International Symposium on* (pp. 1–6). IEEE.

Yan, S., Liao, Y., Feng, X., & Liu, Y. (2016). Real time activity recognition on streaming sensor data for smart environments. In *Progress in Informatics and Computing (PIC), 2016 International Conference on* (pp. 51–55). IEEE.

Yin, J., Yang, Q., & Pan, J. J. (2008). Sensor-based abnormal human-activity detection. *IEEE Transactions on Knowledge and Data Engineering*, *20*, 1082–1090.

Zadeh, L. A. (1996). Fuzzy sets. In *Fuzzy Sets, Fuzzy Logic, And Fuzzy Systems: Selected Papers by Lotfi A Zadeh* (pp. 394–432). World Scientific.