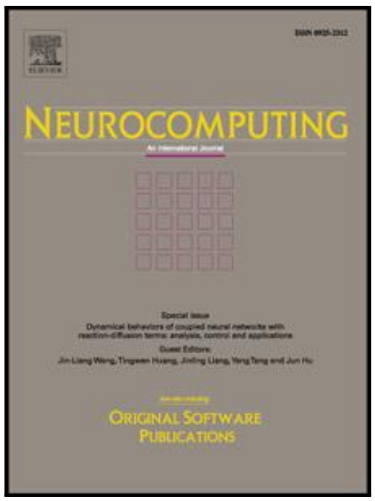


A Sequential Deep Learning Application for Recognising Human Activities in Smart Homes

Daniele Liciotti, Michele Bernardini, Luca Romeo, Emanuele Frontoni

PII: S0925-2312(19)30486-2
DOI: https://doi.org/10.1016/j.neucom.2018.10.104
Reference: NEUCOM 20680

To appear in: *Neurocomputing*

Received date: 31 March 2018
Revised date: 8 September 2018
Accepted date: 12 October 2018

Please cite this article as: Daniele Liciotti, Michele Bernardini, Luca Romeo, Emanuele Frontoni, A Sequential Deep Learning Application for Recognising Human Activities in Smart Homes, *Neurocomputing* (2019), doi: https://doi.org/10.1016/j.neucom.2018.10.104

A Sequential Deep Learning Application for Recognising Human Activities in Smart Homes

Daniele Liciotti[a,*], Michele Bernardini[a], Luca Romeo[a], Emanuele Frontoni[a]

[a]*Department of Information Engineering (DII), Università Politecnica delle Marche, Via Brecce Bianche 12, 60131 Ancona (Italy)*

Abstract

The recent advancement and development of computer electronic devices has led to the adoption of smart home sensing systems, stimulating the demand for associated products and services. Accordingly, the increasingly large amount of data calls the machine learning (ML) field for automatic recognition of human behaviour. In this work, different deep learning (DL) models that learn to classify human activities were proposed. In particular, the long short-term memory (LSTM) was applied for modelling spatio-temporal sequences acquired by smart home sensors. Experimental results performed on the Center for Advanced Studies in Adaptive Systems datasets show that the proposed LSTM-based approaches outperform existing DL and ML methods, giving superior results compared to the existing literature.

Keywords: Smart Home; Human Activity Recognition; Deep Learning; LSTM.

1. Introduction

In the last few decades, human activity recognition (HAR) has been a lively and challenging research area, due to its applicability to different active and assisted living (AAL) domains, as well as the increasing demand for home automation and convenience services for the elderly [1]. Nowadays, mainly because of the rapid increase in the world's ageing population [2], HAR has acquired much interest in the field of ambi-

*Corresponding author

Email addresses: d.liciotti@pm.univpm.it (Daniele Liciotti), m.bernardini@pm.univpm.it (Michele Bernardini), l.romeo@univpm.it (Luca Romeo), e.frontoni@univpm.it (Emanuele Frontoni)

ent intelligence and assisted living technologies in smart homes. It is meant to improve the residents' quality of life with the use of simple and ubiquitous sensors [3]. According to [4], a smart home provides independence and comfort to the residents by using all technological devices interconnected within the network, capable of communicating and learning through the user's habits, creating an interactive space. In particular, HAR is the most salient process for incorporating ambient intelligence into smart environments. It involves a series of complex modelling, reasoning, and decision-making procedures [5, 6]. The goal of HAR is to detect and then identify simple and complex human activities in real-world settings by processing spatial and temporal information acquired by visual and non-visual sensory data [6, 7]. The adopted sensors may be fused in the environment, connected with its objects, or worn directly by the occupant. Compared to wearable sensors, object or environment sensors are advantageous, as they can give an indirect indication of the occupant's activities; moreover, they can discriminate similar actions [3, 8]. According to [7, 9], HAR application domains are among the most varied, but they can be enclosed in three macro categories: healthcare monitoring applications [10], monitoring and surveillance systems for indoor and outdoor activities [11], and lastly, AAL systems [12] for smart homes.

AAL systems are to provide an adequate, non-invasive, technological support, allowing the inhabitants to live independently in safety and in comfort for as long as possible in their homes. Achievement of this goal depends on how the HAR system is able to learn the person's behaviour during daily life. However, the real-world settings are complex and full of uncertainties: data captured by sensors may be ambiguous, as well as noisy and sparse. This leads to the design and implementation of consistent machine learning (ML) techniques to discover knowledge from data and provide a reliable prediction of human behaviour [13]. In order to manage uncertainties and temporal information, data-driven approaches require very large datasets to learn human activities and behaviours [14]. Unfortunately, large real-world datasets are rarely available, and this limitation is one of the major challenges in the AAL field. Accordingly, the knowledge-driven approaches are easy to apply, but they are less robust for managing noisy and temporal data [15]. Hence, it is common practice to consider the HAR task a classification problem. In the past, various classification algorithms have been em-

ployed, such as naive bayes (NB) [16], random forest (RF) [17], hidden markov model (HMM) [16], conditional random field (CRF) [16], k-nearest neighbour (k-NN) [18], and support vector machine (SVM) [19]. Most existing ML approaches result in static models, without the need of evolving and adapting with the changing environment. However [20, 21] proposed an automatic context management system which is able to discover dynamically contextual information, incorporate new data sources, and identify the context with greater discriminative power. Moreover, the self-verification and the reliability measure of the HAR prediction can be performed automatically, measuring the confidence score based on posterior probability and clustering approaches [22].

Each ML algorithm for the HAR task has its own advantages, but none of these approaches prevail over others in all application scenarios [3]. The authors in [23] developed a framework for smart home datasets analysis, employing also the artificial neural network with one hidden layer. Their analysis suggested that the performance of different classifiers is influenced by the dataset characteristic.

For instance, the NB probabilistic classifier obtains good accuracy with large amounts of sample data but does not fit any temporal information. Instead, the HMM and CRF are the most popular approaches for inclusion of such temporal information. Other approaches [24] include hierarchical methods for modelling high-level features which encapsulate past information. Therefore, few existing sensor-based HAR approaches comprehensively explore the temporal patterns among actions [25], handling sequential, interleaved, and concurrent temporal relations [26, 27].

In recent years, there has been an increasing interest in DL techniques for HAR applications [13] and human pose recognition [28–30], which are able to learn multiple, non-linear representations of raw data through multiple hidden layers [31]. This allows the DL application to perform a feature extraction and transformation without prior knowledge. The deep neural network (DNN), convolutional neural network (CNN), recurrent neural network (RNN), and long short-term memory (LSTM) represent the most popular DL techniques in HAR [32–34]. In particular, CNN has been widely used to address the human pose recognition [28] and the HAR task [35] by using convolutions across two or three dimensions in order to capture an image's spatial patterns. Recent research trends in CNN aim to learn the optimal activation functions in a data-

driven way [36] while learning a Euclidean embedding [37] of the feature's space.

In this work, we propose a novel application of LSTM networks to improve HAR within a sensor-fused AAL scenario. We started from the unidirectional LSTM (Uni-LSTM), and we explored more complex LSTM architectures, such as the bidirectional LSTM (Bi-LSTM) and cascade bidirectional and unidirectional LSTM (Casc-LSTM). Two ensemble LSTM approaches named Ensemble2LSTM (Ens2-LSTM) and CascadeEnsemble (CascEns-LSTM) are also proposed. Unlike other DL approaches for video- [13] and wearable-based HAR [9, 38, 39], this paper contextualizes the problem in the smart home scenario where a typical home is equipped with several sensors and the captured data is voluminous and structurally rich [16]. Moreover, the proposed setting is *(i)* closest to the real-world situation [3], *(ii)* privacy compliant [3], and *(iii)* based on an HAR task that is much more challenging due to the variability of activities and residents [16] (e.g., some residents may be younger adults, healthy older adults, or older adults with pathological conditions, and some may perform interactions with pets). Two conditions motivate this particular application of the LSTM network: it allows *(i)* extracting highly discriminative non-linear feature representations while *(ii)* modelling temporal sequences by learning long-term dependency situations. This is often the case with human activities, where the action can be divided into a sequence of gestures and postures, and each sample can be related to the previous ones [40]. The reliability of the proposed approach for HAR is investigated in a smart home scenario using the widely spread Center for Advanced Studies in Adaptive Systems (CASAS) benchmark datasets [41]. Even though the CASAS datasets are widely used and investigated by researchers using supervised ML algorithms, to the best of the authors' knowledge, there is still a lack of HAR works related to the use of DL approaches which take into account the temporal information.

The overall performance of the proposed LSTM-based approaches has been compared with one-dimensional CNN [38]) and traditional ML techniques (e.g., NB, HMM, CFR) widely used in literature for HAR [16].

Results of this study show that the application of standard LSTM leads to significant performance improvement with respect to ML approaches (from 10.60% to 16.04% of accuracy improvement) and with respect to one-dimensional CNN (from 11.03% to

18.39% of accuracy improvement).

2. Related Work

Taking all of these facts into account, it may be useful to fill this gap by applying an adaptive, well-known DL model (i.e., LSTM) on a "rarely been used" dataset in this context. Likewise, it seems challenging to compare the results obtained with other DL and traditional ML models largely present and already discussed in the literature.

2.1. CASAS datasets background

This section focuses on the HAR approaches where the CASAS datasets were employed. In this context, several ML techniques were applied and tested to solve the activity recognition task.

The authors in [16] applied three ML algorithms (i.e., NB, CRF, and HMM) for HAR in the 11 labelled CASAS datasets. A further comparison with SVM was introduced in [19]. Once the SVM model was determined to perform better than other traditional ML techniques, it had been applied for recognising activities in the real world [19].

The reliability of standard, supervised classifiers varies dramatically between datasets and within the single activities [16]. In particular, the classification accuracy is influenced by *(i)* the amount and nature of training data, *(ii)* the ambiguity of the label, and *(iii)* the number of residents investigated at the same time. Different approaches have tried to overcome these crucial issues. In [42], the authors developed a supervised behaviour classification model (BCM) derived from an SVM classifier to differentiate a person from an inhabitant group. Considering only the early morning routine, the BCM extracted features and interpreted the temporal sequence of all users' information captured by sensors. The multiple kernel SVM approach was applied in [43] for the recognition of individual activities. Additionally, the CRF was used to discover sequential future behaviour patterns. In [18], the authors proposed an activity recognition approach by clustering-based classification. They combined the k-NN with the Dempster–Shafer theory of evidence to discriminate and split activity instances of different classes enclosed inside a unique cluster. A Markov logic network was designed

in [44], in order to classify the ongoing activity through probabilistic reasoning. This hybrid segmentation approach can automatically segment continuous and sparse sensor events into discrete sequences, ensuring a correct interpretation of the input raw data that the sensors generated.

The key finding of several state-of-the-art approaches is to model the action as a sequence of subsequent gestures/behaviours over time. This leads to recognition of time-dependent patterns of events, predicting future behaviours starting from the current activity or state.

In [45], the authors proposed an activity-prediction model using probabilistic Bayesian networks and a novel two-step inference process to predict (i) the next activity and (ii) its related start time. In [46], the authors tried to estimate prior probabilities of an activity happening at a certain time, in order to reduce the error rate of a given classification algorithm. Several temporal models such as frequency map enhancement and Gaussian mixtures model were evaluated. The time relevance was analysed in [17], where the authors proposed a time–space feature importance analysis in order to compare the potential relevance of features for activities classification. RF, NB, and SVM were the adopted techniques used for discriminating the feature relevance. In [47], the authors presented an activity forecasting method that can predict the expected time until an activity occurs. This method generates an activity forecast using a regression tree classifier and offers an advantage over sequence prediction methods, such as linear regression and SVM classifiers.

DL algorithms for HAR applied to CASAS datasets have still been scarcely explored in the literature. The authors in [48] implemented a deep belief network comprising many restricted Boltzmann machines. Then, the performance of the proposed network was compared with other traditional ML algorithms such as HMM, NB, and CRF, which were also employed in [16], showing interesting results.

Unlike the above-mentioned work, this paper proposes the application of a sequential DL LSTM approach for HAR. This leads to some advantages with respect to traditional ML models:

- LSTM allows automatic learning of spatio-temporal information from the sensor

data without the need of handcrafted features [49] or kernel fusion approaches [43];

- LSTM, as a sequential approach, models the temporal evolution of the features, using recurrent connections in the hidden layers.

2.2. DL for HAR using time series data

DL approaches were employed to capture the temporal dependency of a human action, considering time series data [13]. In particular, CNNs adopt convolutions across a one-dimensional temporal sequence to capture local dependencies among input data, using parameter sharing across time [38]. However, the geometry of convolutional kernels restricts the captured range of dependencies between data samples; likewise, local connectivity limits the output to a function of a small number of neighbouring input samples [39]. As a result, CNNs may be unsuitable to a wide range of HAR configurations and require fixed-length input windows. The use of LSTM may overcome these limitations by exploiting their internal memories to capture long-range dependencies in variable-length input sequences. As we shall see in the Results section, our LSTM-based approaches perform favorably over CNN [38].

Moreover, in accordance with the most recent state-of-the-art contributions [13], ensemble DL models, RNN, and LSTM have not yet been well investigated for HAR using multimodal channels. Specifically, Bi-LSTM and Casc-LSTM models have already been employed to recognise human activity using wearable and on-body sensor input data [39]. However, deviating from [39], we aim to explore an LSTM-based methodology for HAR using only a sequence of data acquired by domotic fused sensors (e.g., motion sensors, temperature sensors, magnetic door sensors). The proposed multimodal data configuration is more challenging than wearable sensor data are, because smart home data do not always display a discriminative pattern, due to the intrinsic variability of activities and residents [16].

3. Material and Methods

3.1. Smart home datasets

Several datasets were used by researchers in a smart home scenario for HAR applications [50–54]. Since the collection of real house data is costly, time consuming,

and difficult to obtain, the publicly available datasets have a crucial importance for the research community. Additionally, they are useful for testing HAR algorithms and providing the baseline for comparison. In Table 1, a brief overview of the widely used, publicly available smart home datasets is reported.

Table 1: Smart home datasets.

Dataset	# Houses	Residents	# Sensors	# Activities
CASAS [41]	7	Multi	20 - 86	11
Kasteren [50]	3	Multi	14 - 21	14 - 16
Domus [51]	1	Single	78	User's feelings
ARAS [52]	2	Multi	20	27
HIS [53]	1	Multi	20 - 30	7
OPPORTUNITY [54]	1	Multi	72	15 - 20

3.2. CASAS datasets

The CASAS datasets were introduced by Washington State University [41]. The testbed smart apartment used in the CASAS smart home project comprised three apartments that include three bedrooms, one bathroom, a kitchen, and a living/dining room. Each apartment was equipped with different kinds of sensors (e.g., motion sensors, temperature sensors, door sensors) and actuators for sensing the environment and providing information to inhabitants. Five annotated datasets, named Milan, Cairo, Kyoto2, Kyoto3, and Kyoto4 were selected among all available CASAS datasets [55]. This choice was motivated by the fact that these datasets present the same sensor data representation, defined as date-time, sensor, and state/value (see Table 2).

- The Milan dataset contains sensor data collected in the home of a volunteer adult. The residents were a woman and a dog. The woman's children visited on several occasions. The sensor events were generated from motion (M), door closure (D), and temperature (T) sensors;
- the Cairo dataset contains sensor data collected in the home of a volunteer adult couple. The residents were a man, a woman, and a dog. The couple's children also visited the home on at least one occasion. The sensor events were generated from M and T sensors;

Table 2: Example of sensors data representation.

Date-Time	Sensor	State/Value
02/02/2009 12:18:44	M16	On
02/02/2009 12:18:46	M17	Off
02/02/2009 12:28:50	D12	Open
02/02/2009 12:29:55	I03	Present
05/02/2009 08:05:52	AD1-B	0.0448835
05/02/2009 12:21:51	D09	Closed
10/02/2009 17:03:57	I03	Absent
⋮	⋮	⋮

- the Kyoto datasets contain sensor data collected in an apartment that housed two residents, R1 and R2, when they performed normal daily activities. The sensor layout comprised M, D, and T sensors, a burner sensor (AD1-A), a hot water sensor (AD1-B), a cold water sensor (AD1-C), an item sensor (I) for selected items in the kitchen, and an electricity usage sensor (P001).

Table 3 shows the main information of these datasets, including the number of activities as well as number of occurrences, the type and the number of sensors, the number of residents, and the number of monitored days.

Table 3: CASAS datasets.

CASAS Dataset	Milan	Cairo	Kyoto2	Kyoto3	Kyoto4
Residents	1+pet	2+pet	2	3	3
Sensors	32	27	71	86	72
Type of sensor	M,T,D	M,T	M,T,D,I,P001 AD1-A/B/C	M,T,D,I,P001 AD1-A/B/C	M,T,D,I,P001 AD1-A/B/C
Activities	15	13	13	12	25
Activity occurrences	1513	600	497	1342	844
Days	92	56	46	64	250

The different activities for each dataset are summarized in Table 4. According to the configuration used in [16], the original activities listed below for each dataset have been grouped into 11 activities of daily living (ADL) to make a consistent comparison of the results. Moreover, this choice is motivated by the fact that the selected activities occur in the majority of the investigated datasets and are typically used to discriminate the functional health of an individual within a clinical scenario. Therefore, the chosen activity classes are as follows: *Personal hygiene*, *Sleep*, *Bed to toilet*, *Eat*, *Cook*, *Work*, *Leave home*, *Enter home*, *Relax*, *Take medicine*, and *Bathing*. Activities not represented by the previous categories, as well as activities named "*None*", in which the resident performed no activity, was grouped as "*Other*".

3.3. Proposed LSTM models

The proposed HAR framework comprised the preprocessing and classification stage (see Figure 1). The preprocessing stage included the filtering and data aggregation from the CASAS datasets. The input features were the raw data collected from different smart home sensors (e.g., M, D, T). Considering the time lapse from the beginning to the end of the human activity, the data aggregation was performed in order to encapsulate all changes in sensor status. This processing led to associate an input matrix of sensor events for each activity. The input data was fed into the LSTM-based model in order to predict the class-membership. A brief background of the LSTM model is provided in Section 3.3.1. Afterwards, we investigated more details of the LSTM by not only implementing and testing the standard Uni-LSTM (see Section 3.3.2) but also considering more complex architectures such as Bi-LSTM (see Section 3.3.3) and Casc-LSTM (see Section 3.3.4). Moreover, we presented two different ensemble strategies named Ens2-LSTM and CascEns-LSTM. The former combined the output of a Bi-LSTM and LSTM (see Section 3.3.5), while in the latter, the output of the Ens2-LSTM was fed into an LSTM (see Section 3.3.6). These ensemble strategies aim to improve the generalisation performance while learning more complex patterns from data.

Table 4: ADLs of the tested CASAS datasets.

	Milan	**Cairo**	**Kyoto2**	**Kyoto3**	**Kyoto4**
Bathing	Master Bathroom Guest Bathroom	-	-	R1 Shower R2 Shower	R1 Bathing R2 Bathing
Bed to toilet	Bed to Toilet	Bed to Toilet	R1 Bed to Toilet R2 Bed to Toilet	Bed to Toilet	R1 Bed to Toilet R2 Bed to Toilet
Cook	Kitchen Activity	-	Meal Preparation	Cooking	R1 Meal Preparation R2 Meal Preparation
Eat	Dining Rm Activity	Breakfast Dinner Lunch	-	-	R1 Eating R2 Eating
Enter home	-	-	-	-	R1 Enter Home R2 Enter Home
Leave home	Leave Home	Leave Home	-	-	R1 Leave Home R2 Leave Home
Personal hygiene	-	-	R1 Personal Hygiene R2 Personal Hygiene	-	R1 Personal Hygiene R2 Personal Hygiene
Relax	Read Watch TV	-	Watch TV	-	R1 Watch TV R2 Watch TV
Sleep	Sleep	R1 Sleep R2 Sleep	R1 Sleep R2 Sleep	R1 Sleep R2 Sleep	R1 Sleep R1 Sleeping Not in Bed R2 Sleep R2 Sleeping Not in Bed
Take medicine	Morning Meds Evening Meds	R2 Take medicine	-	-	-
Work	Chores Desk Activity	R1 Work in Office Laundry	Clean R1 Work R2 Work	Cleaning R1 Work R2 Work	R1 Work R2 Work R1 Housekeeping
Other	Master Bedroom Meditate	R1 Wake Night Wandering R2 Wake	Study Wash Bathtub	Grooming R1 Wake R2 Wake	R1 Wandering in Room R2 Wandering in Room

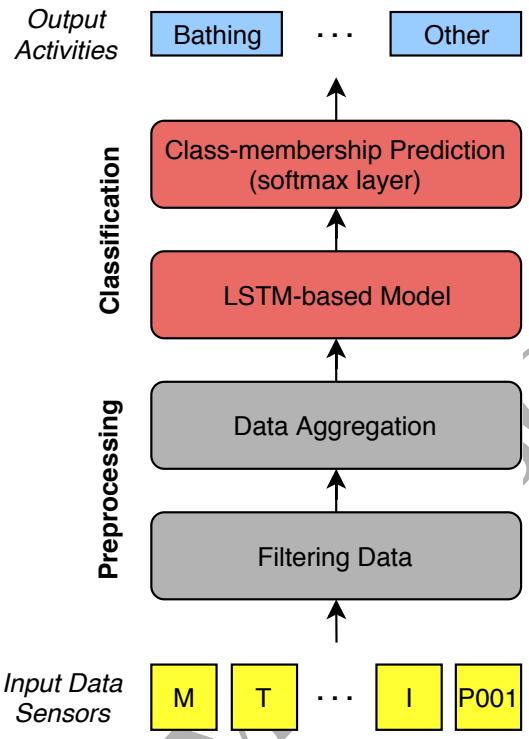


Figure 1: The proposed HAR approach. Different data modalities acquired by different sensors are fed into the LSTM-based model in order to predict human activity.

3.3.1. *LSTM background*

LSTM networks [56] can be seen as a very successful extension of the RNN, explicitly designed to avoid the long-term dependency problem associated with RNNs. In particular, [57] demonstrated that the RNN can model the short time-lags between input and labels. However, this short-term memory can be insufficient when dealing with real-world time series data [57]. LSTM methodology proposed a special node, called the constant error carousel (CEC), that allows constant propagation of the error signal over time. Additionally, it uses the gating mechanism over an internal memory cell to control access to the CEC and to learn and represent a more complex representation of the long-term dependencies. Hence, an LSTM model is well suited to classify, process, and predict time series with time lags of unknown sizes.

The LSTM layer's main component is the memory cell. The cell is responsible for "remembering" values or states for long or short time periods over arbitrary time intervals. An LSTM block usually contains input, output, and forget gates, which are respectively, seen as write, read, and reset operations for the memory cell. Each of the three gates can be thought of as a "conventional" artificial neuron, as in a multi-layer neural network. Thus, using particular activation functions, gates can regulate the flow of values that go through the connections of the LSTM layer. An LSTM cell state is the key component which carries the information between each LSTM block. Modifications to the cell state are controlled with the three gates described above. The single cell, as well as the gates, are interconnected and connected to the cell state itself.

3.3.2. Uni-LSTM

In order to classify the action time series, the use of an RNN architecture with one hidden layer of LSTM cells is proposed. The input layer of this RNN comprises an embedded vector that contains the sequence of sensor events. Then, n LSTM cells are fully connected to these inputs and have recurrent connections with all the LSTM cells. A dense output layer performs the classification task. The number of cells (n) and the learning rate are the common hyperparameters for all LSTM-based approaches selected in the validation procedure. The RMSProp optimiser [58] was used for training the network and minimising the categorical cross entropy loss function.

Figure 2 shows the LSTM single cells over time. The single cell layer is presented at time t, where X_t and Y_t are the input and output states, respectively. Data from each sensor reported in Table 2 represent the input, while the different activities for each dataset represent the output (see Table 4).

3.3.3. Bi-LSTM

The Bi-LSTM [59] includes two parallel LSTM tracks defined by forward and backward loops, which extract patterns from the past and the future, in order to better model time dependency (see Figure 3). The forward track (green arrow) reads the input data X_t from left to right, whereas the backward track (red arrow) reads the input data from right to left. The output prediction is the weighted sum of the prediction score,

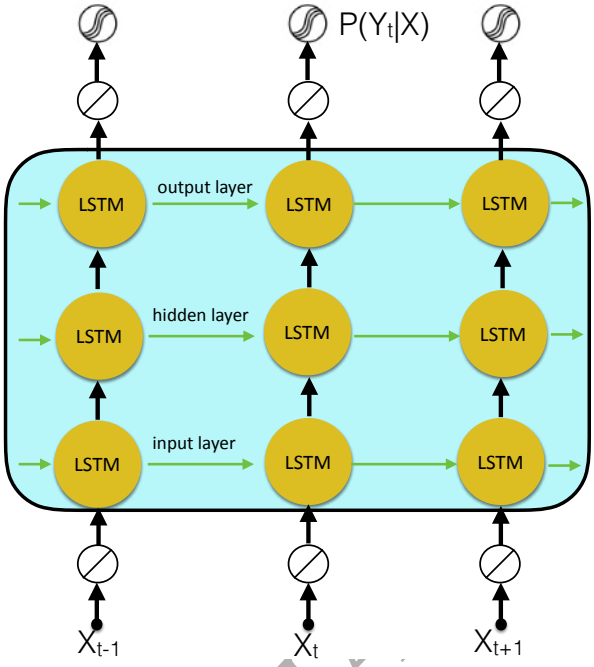


Figure 2: Overview of the unidirectional LSTM (Uni-LSTM) architecture, comprising one input, hidden, and output layer: X represents the binary vector for sensor inputs, and Y represents the activity label prediction of the LSTM network.

resulting from the forward and backward tracks [59].

3.3.4. *Casc-LSTM*

The cascade architecture is inspired by [60]. The input layer is a Bi-LSTM cascaded with the Uni-LSTM. Thus, the output of one-layer Bi-LSTM is considered as the features vector to feed into the Uni-LSTM (see Figure 4).

3.3.5. *Ens2-LSTM*

The Ens2-LSTM aims to combine the output of a Bi-LSTM and Uni-LSTM. In particular, the softmax function combines the output of the two models in order to predict human activity (see Figure 5).

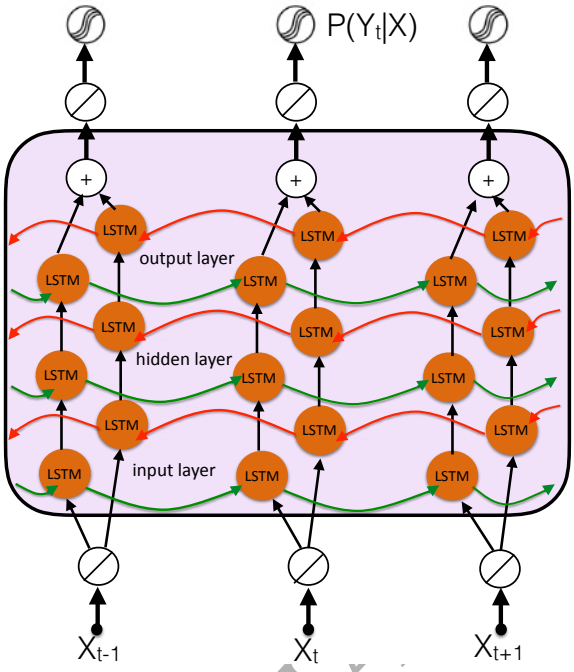


Figure 3: Bidirectional LSTM-based (Bi-LSTM) architecture [59], comprising one input, hidden, and output layer. The forward and backward tracks are defined for each layer.

3.3.6. CascEns-LSTM

The CascEns-LSTM is the cascade of the Ens2-LSTM and Uni-LSTM. In particular, the Ens2-LSTM prediction represents the input vector to feed into the Uni-LSTM.

4. Experimental

The same experimental setup employed in [16] was designed (see Figure 7) in order to perform a fair comparison between our LSTM approach and the HMM, CRF, and NB approaches. Hence, a stratified (over class) threefold cross-validation procedure was performed, and the accuracy result is the average of all folds. The hyperparameters were optimised with a hold-out procedure. Twenty percent of the training data was used as validation data and was not considered for training the model. The sparse categorical cross-entropy loss was evaluated in order to select the best hyperparameters.

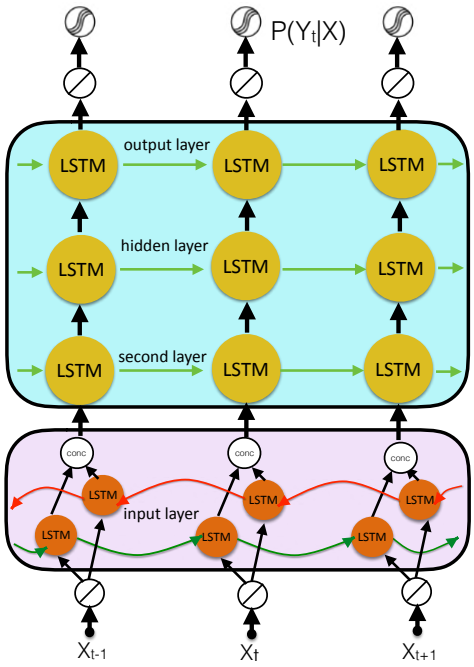


Figure 4: Cascaded bidirectional and unidirectional LSTM-based (Casc-LSTM) architecture [60]. The upper layers are unidirectional, whereas the input layer is bidirectional. We set one hidden unidirectional layer.

Different configurations in the number of units (i.e., 32, 64, and 128) and the learning rate were considered as common hyperparameters of the LSTM-based models. Hence, a configuration of $n = 64$ was found to be the optimal compromise to avoid overfitting and achieving a low generalisation error. Based on the analysis presented by [16], the Bosch and Kyoto1 datasets were not considered: the Bosch datasets were excluded because they are not publicly available, while the Kyoto1 dataset has a limited number of occurrences/samples.

5. Results

Figure 8 reports the training and validation accuracy of Uni-LSTM for different numbers of units (i.e., $n = 32$, $n = 64$, and $n = 128$). The lowest generalisation error in the validation set was achieved with $n = 64$.

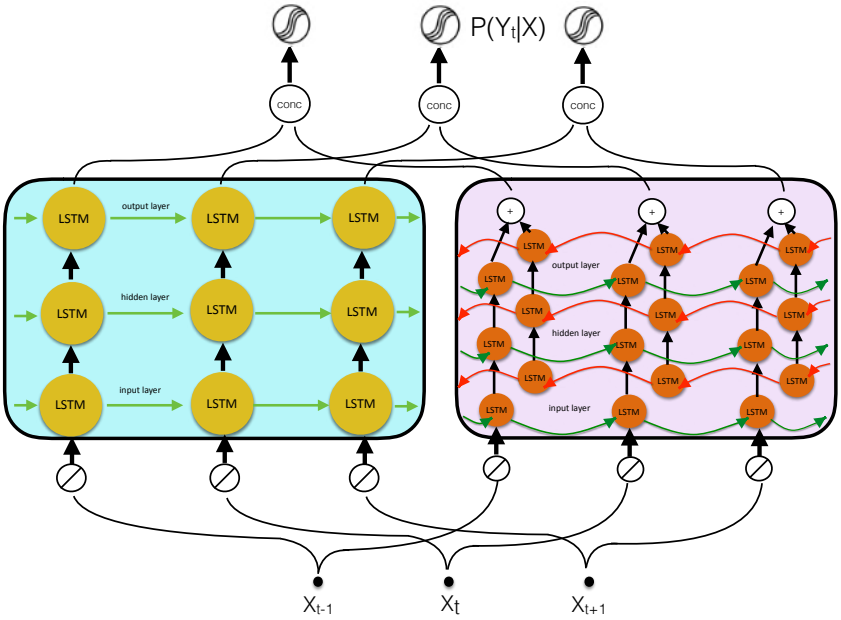


Figure 5: Ensemble2LSTM (Ens2-LSTM) architecture. The unidirectional and bidirectional models comprise of one input, hidden, and output layer.

Figure 9 shows the averaged accuracy of each fold and the related standard deviation for all the LSTM-based models. They achieved an accuracy of above 65% and significantly higher-than-chance level (i.e., 1/number of classes). Even if each LSTM model, except Casc-LSTM for Cairo, Kyoto2, and Kyoto3, obtained very similar results, the Bi-LSTM was the most effective methodology for the Cairo and Kyoto2 datasets. However, the Ens2-LSTM achieved the best accuracy for the largest datasets, in terms of number of observations for each class (e.g., Milan, Kyoto3, and Kyoto4).

The overall results of the LSTM approaches are also reported in Table 5, in terms of averaged precision, recall, and f1-score. Although all datasets are highly unbalanced, the precision, recall, and f1-score follow the same trend of accuracy.

Figure 10 shows the confusion matrices for the Milan dataset, obtained by each different LSTM model. Although most of the considered activities were correctly classified, the most relevant errors can be summarised as follows: *(i) Bed to toilet* as *Bathing*

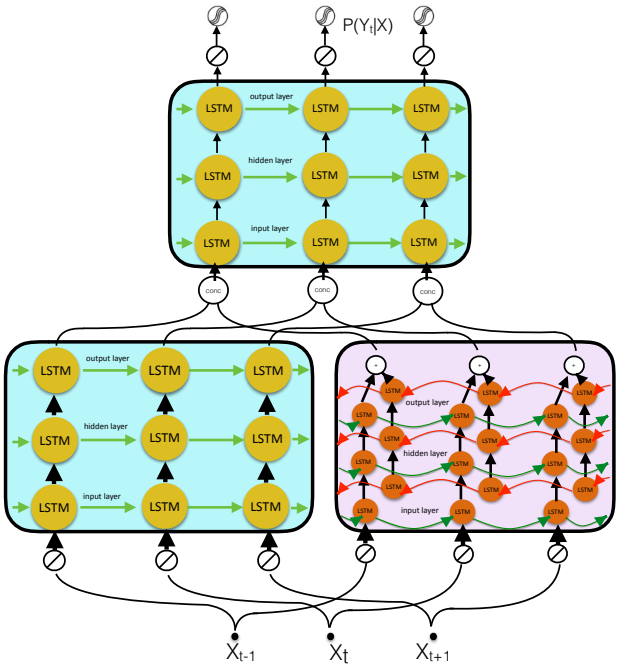


Figure 6: CascadeEnsemble (CascEns-LSTM) architecture. The unidirectional and bidirectional models comprise one input, hidden, and output layer.

and *(ii) Eat* as *Other*. In case *(i)*, the misclassification occurred in a borderline situation when two similar activities were executed approximately in the same area; in case *(ii)*, the misclassification may have been due to the imbalance in the number of occurrences between the two activities (i.e., generally a higher number of observations of *Other* with respect to other classes).

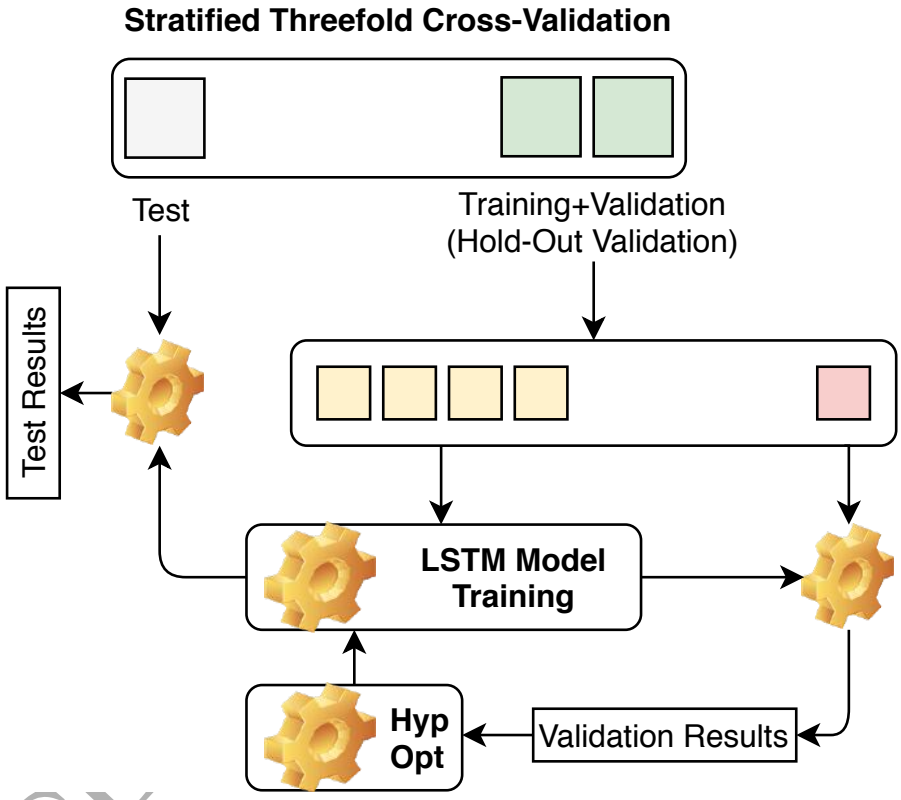


Figure 7: Classification stage.

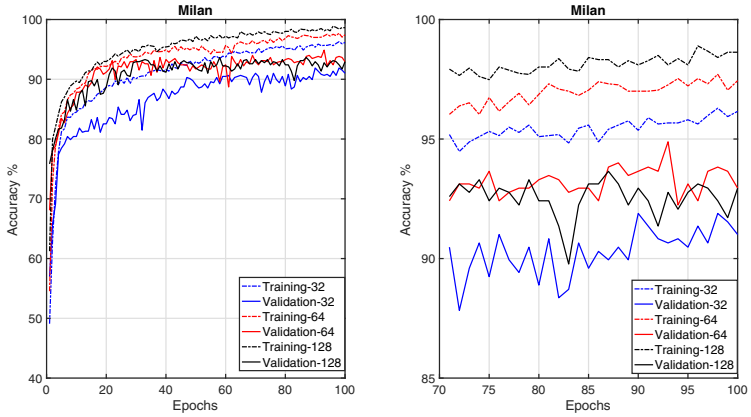


Figure 8: Training and validation accuracy with $n = 32$, $n = 64$, and $n = 128$. The right side of the figure represents a zoomed overview of the left one.

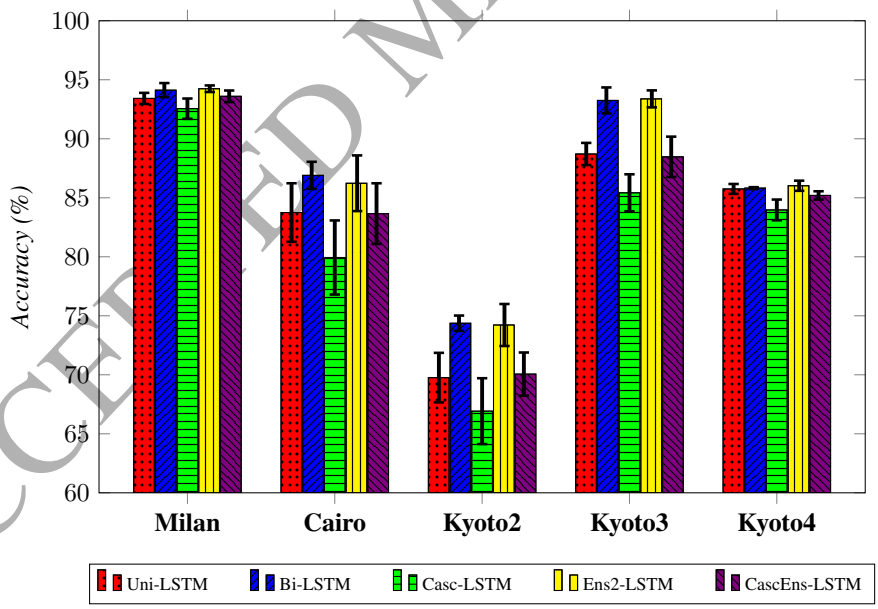
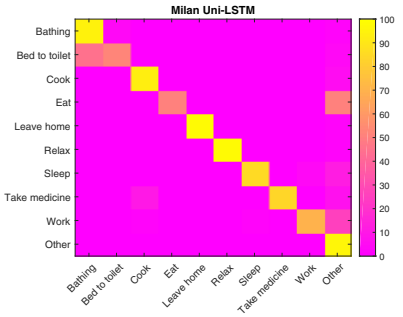


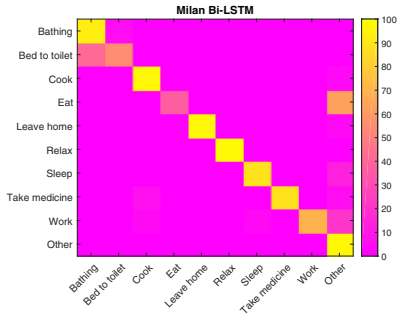
Figure 9: Averaged accuracy and standard deviation of each dataset over all three folds.

Table 5: LSTM results: accuracy, precision, recall, and f1-score for each of the five datasets.

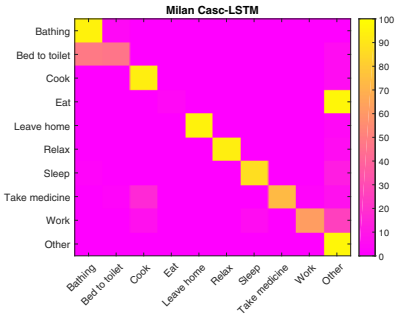
Metric	**Model**	**Dataset**				
		Milan	Cairo	Kyoto2	Kyoto3	Kyoto4
Accuracy (%)	LSTM	93.42	83.75	69.76	88.71	85.57
	Bi-LSTM	94.12	**86.90**	**74.37**	93.25	85.82
	Casc-LSTM	92.55	79.94	66.92	85.42	83.97
	Ens2-LSTM	**94.24**	86.23	74.22	**93.38**	**86.02**
	CascEns-LSTM	93.60	83.66	70.06	88.47	85.20
Precision (%)	LSTM	93.67	83.33	70.00	88.67	85.67
	Bi-LSTM	94.00	**86.67**	**75.00**	93.33	86.00
	Casc-LSTM	92.00	79.33	68.33	85.67	83.67
	Ens2-LSTM	**94.33**	86.33	74.67	**93.67**	**86.33**
	CascEns-LSTM	93.33	84.33	70.33	86.67	85.00
Recall (%)	LSTM	93.67	82.33	71.00	88.33	85.33
	Bi-LSTM	94.00	**87.00**	**74.33**	**93.33**	**86.00**
	Casc-LSTM	92.67	80.00	67.00	85.67	84.00
	Ens2-LSTM	**94.33**	86.33	**74.33**	**93.33**	**86.00**
	CascEns-LSTM	93.33	84.00	70.33	88.00	85.33
f1-score (%)	LSTM	93.33	83.33	69.67	88.33	85.33
	Bi-LSTM	**94.00**	**86.67**	**74.33**	**93.33**	**86.00**
	Casc-LSTM	92.00	78.67	66.00	85.33	83.33
	Ens2-LSTM	**94.00**	86.00	73.67	**93.33**	**86.00**
	CascEns-LSTM	93.33	83.67	69.67	88.33	85.00



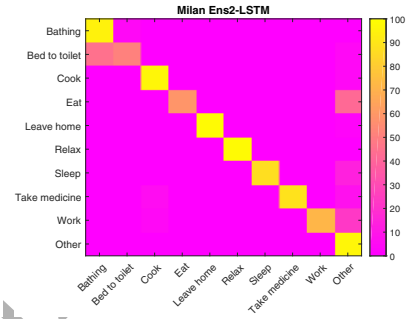
(a) Uni-LSTM



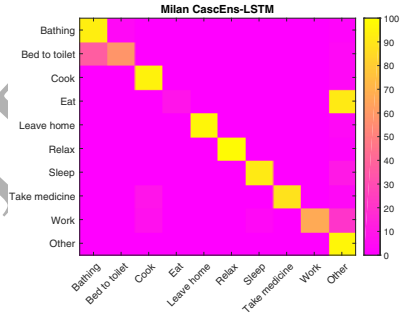
(b) Bi-LSTM



(c) Casc-LSTM



(d) Ens2-LSTM



(e) CascEns-LSTM

Figure 10: Confusion matrices of the Milan dataset for Uni-LSTM (Figure 10a), Bi-LSTM (Figure 10b), Casc-LSTM (Figure 10c), Ens2-LSTM (Figure 10d), and CascEns-LSTM (Figure 10e) approaches: rows are the true classes; columns are the predicted ones. Below, the percentage number of occurrences for each class is reported: *Bathing* = 14.95%, *Bed to toilet* = 2.09%, *Cook* = 13.03%, *Eat* = 0.52%, *Leave home* = 5.03%, *Relax* = 10.06%, *Sleep* = 2.26%, *Take medicine* = 1.41%, *Work* = 1.81%, and *Other* = 48.84%.

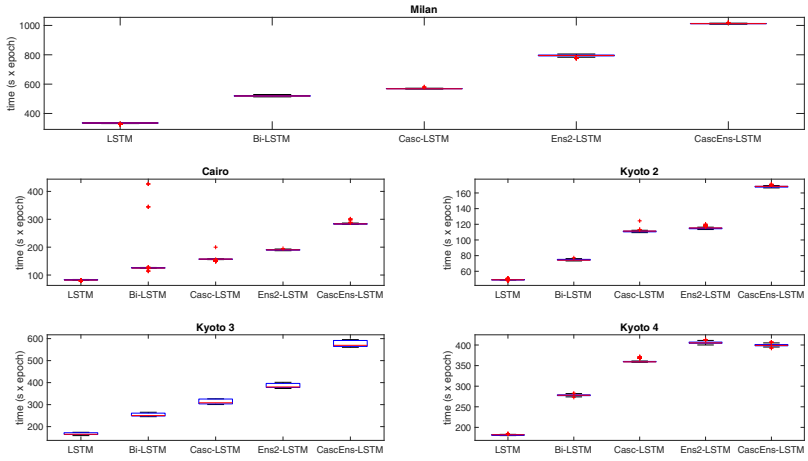


Figure 11: Box plot computation time training phase (s x epoch) for all datasets and all LSTM-based methodologies.

5.1. Computation time

Figure 11 shows the box plot of the computation time, for each epoch, for the training stage of all datasets. All the experiments are reproducible, and they were performed using Intel Core i7-4790 CPU 3.60GHz with 16GB of RAM and NVIDIA GeForce GTX 970.

The training stage of the Uni-LSTM was faster than that of the other LSTM-based methodologies. The computation time increases when the complexity of the model grows.

Figure 12 shows the computation time for testing the LSTM-based methodologies, considering the five employed datasets. The testing time was averaged over the three-fold cross-validation.

5.2. Comparison with other DL approaches

The proposed LSTM-based methodologies were compared with respect to one-dimensional CNN, widely employed for HAR, using multimodal time series data [31, 38].

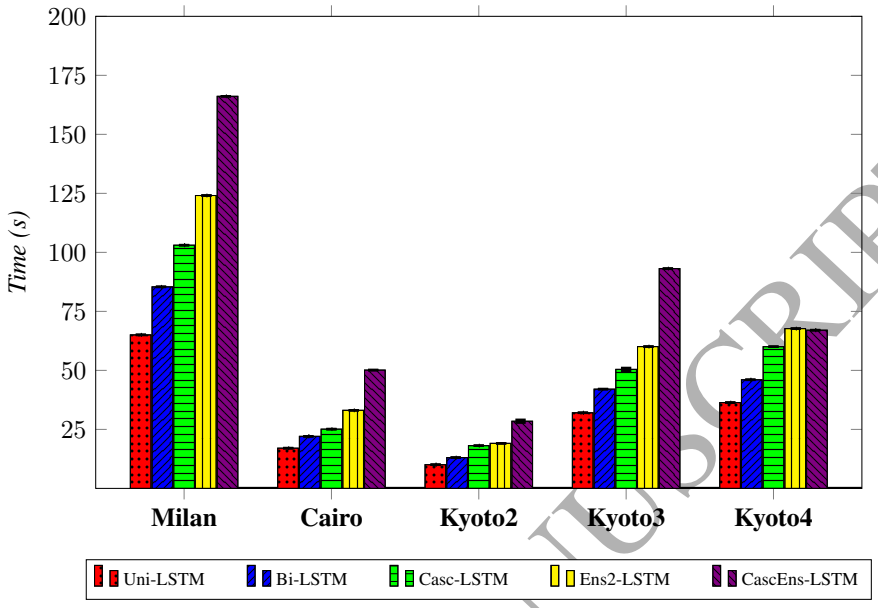


Figure 12: Computation time testing phase averaged over the threefold cross-validation.

The CNN architecture encapsulates at least one temporal convolution layer, one pooling layer, and at least one fully connected layer, followed by a top-level softmax group [38].

We regularised the CNN to avoid overfitting. In particular, we applied a dropout strategy after each max-pooling or fully connected layer, according to [38], and a max-in norm regularisation as suggested in [61]. Thus, we performed three experimental tests:

1. CNN1: the dropout probability of the i-th layer p^i_{drop} fixed to 0.2 for all layers;
2. CNN2: $p^1_{drop} = 0.1$, $p^2_{drop} = 0.25$, $p^{i>2}_{drop} = 0.5$;
3. CNN3: max-in norm regularisation and $p^1_{drop} = 0.1$, $p^2_{drop} = 0.25$, $p^{i>2}_{drop} = 0.5$ [38].

Table 6 shows the CNN results for each dataset, as compared with the standard Uni-LSTM.

The Uni-LSTM overcomes all the CNN-based models for all the considered datasets.

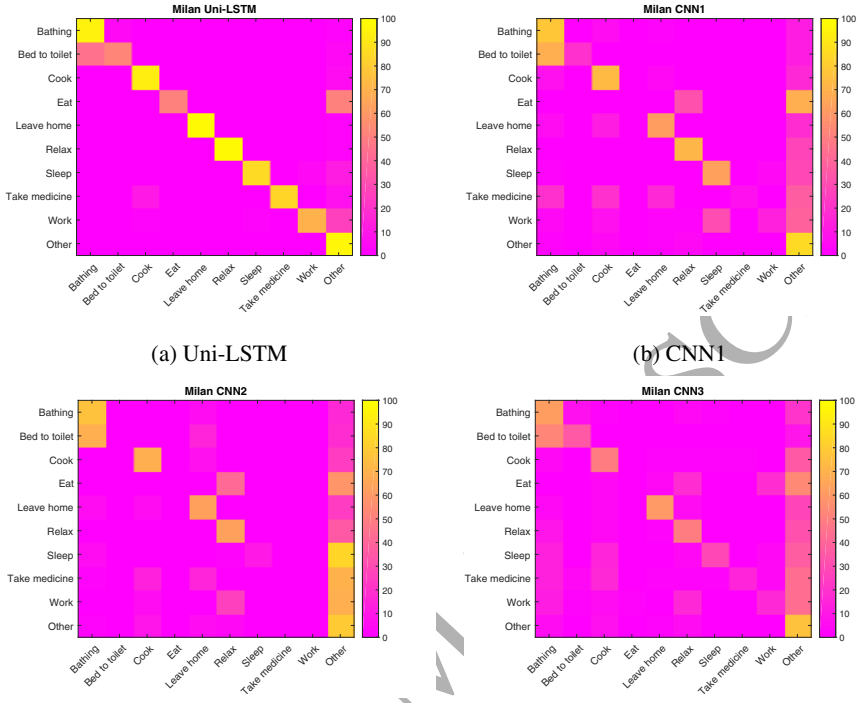
Table 6: CNN results: accuracy, precision, recall, and f1-score for each of the five datasets.

Metric	Model	Dataset				
		Milan	Cairo	Kyoto2	Kyoto3	Kyoto4
Accuracy (%)	Uni-LSTM	**93.42**	**83.75**	**69.76**	**88.71**	**85.57**
	CNN1	75.03	70.67	58.56	77.68	69.39
	CNN2	68.33	74.16	58.54	73.92	61.82
	CNN3	62.50	69.23	49.65	71.93	59.34
Precision (%)	Uni-LSTM	**93.67**	**83.33**	**70.00**	**88.67**	**85.67**
	CNN1	73.67	70.00	57.00	76.33	67.00
	CNN2	64.33	73.33	59.33	71.66	59.33
	CNN3	61.66	66.33	50.33	71.66	58.00
Recall (%)	Uni-LSTM	**93.67**	**83.67**	**70.00**	**88.67**	**85.67**
	CNN1	75.00	70.67	58.67	77.67	69.33
	CNN2	68.33	74.00	58.67	73.67	62.00
	CNN3	62.66	69.33	49.67	72.00	59.33
f1-score (%)	Uni-LSTM	**93.33**	**83.33**	**69.67**	**88.33**	**85.33**
	CNN1	73.66	69.67	57.33	76.67	67.00
	CNN2	65.66	72.66	57.67	72.33	56.67
	CNN3	61.66	67.00	49.33	71.33	58.67

This can be explained by the potential of LSTM to capture long-term temporal dependencies of human action, achieving the best human activity prediction.

Figure 13 shows the confusion matrices of the CNN-based models for the Milan dataset. It is noted that the misclassified activities increase by employing CNN models. Generally, the CNN1 model was less affected by class unbalance and performed better than the CNN2 and CNN3 models for the Milan dataset. In addition to the misclassified activities in common with LSTM approaches (i.e., *Bed to toilet* as *Bathing*, *Eat* as *Other*), CNN models were unable to correctly classify *Take medicine* and *Work*, which

were always confused with the majority class *Other*.



(a) Uni-LSTM (b) CNN1

(c) CNN2 (d) CNN3

Figure 13: Confusion matrices of the Milan dataset for CNN1 (Figure 13b), CNN2 (Figure 13c), and CNN3 (Figure 13d) approaches: rows are the true classes; columns are the predicted ones. Below, the percentage number of occurrences for each class is reported: *Bathing* = 14.95%, *Bed to toilet* = 2.09%, *Cook* = 13.03%, *Eat* = 0.52%, *Leave home* = 5.03%, *Relax* = 10.06%, *Sleep* = 2.26%, *Take medicine* = 1.41%, *Work* = 1.81%, and *Other* = 48.84%.

5.3. Comparison with other ML approaches

Table 7 shows the comparison between the proposed LSTM approach and the ML methods employed in [16].

Table 7: NB, HMM, CRF, and LSTM recognition accuracies for each of the five datasets.

Dataset	NB (%)	HMM (%)	CRF (%)	LSTM (%)
Milan	76.65	77.44	61.01	**93.42**
Cairo	82.79	82.41	68.07	**83.75**
Kyoto2	63.98	65.79	66.20	**69.76**
Kyoto3	77.50	81.67	87.33	**88.71**
Kyoto4	63.27	60.90	58.41	**85.57**

The LSTM model outperformed ML methods for all considered datasets. For Milan and Kyoto4, the LSTM achieved the highest improvement (Milan: 16.77%, 15.98%, 32.41%; Kyoto4: 22.30%, 24.67%, 27.16%) compared to NB, HMM, and CRF.

6. Discussion and Conclusions

The results presented in this paper show that the applied DL approach based on LSTM can lead to a viable solution to improve significantly the ADLs' recognition task in the smart home scenario.

In particular, after a comprehensive comparison with the best recent literature focused on HAR techniques, the LSTM methodology applied to the CASAS datasets evidently outperforms both alternative DL approaches (i.e., DNN, CNN) and existing traditional ML techniques (i.e., NB, HMM, CRF).

Starting from the standard LSTM formulation, we explored more complex LSTM-based models in order to improve the generalisation performance of HAR prediction. However, the increase in complexity did not always lead to a significant improvement of performance. Even if the metrics of all proposed LSTM models (excluding Casc-LSTM for Kyoto2) are very close to each other for every dataset, the best results were achieved by Bi-LSTM and Ens2-LSTM models. This result confirms how future input

information is typically also useful for HAR. The Bi-LSTM exploits all available input information in the past and future of a specific time frame. Moreover, the Ens2-LSTM slightly overcomes the Bi-LSTM in all datasets with the higher number of occurrences (i.e., Milan, Kyoto3, and Kyoto4; see Table 3). However, a unique best performer model did not emerge across the different CASAS datasets, which display several real-life challenges. Even though the nature of the collected data is the same, other factors may have affected the performance of the LSTM models, such as the number of residents, the number and type of sensors, the number of different activities, and the duration of test days. Additionally, the proposed LSTM methodology is consistent with the highly unbalanced setting of the smart home dataset, without requiring data augmentation techniques. Results show that the LSTM approaches are able to better generalise across different samples of the same user. This is the case of the Milan dataset, where there was only one inhabitant. Accordingly, the performance of the LSTM models decreased in the Cairo and Kyoto datasets, where the activities of two/three residents were monitored. This suggests that in future works, the proposed approach has considerable opportunities for improvement and could be easily adapted and implemented for better multi-user activity recognition. Hence, a multi-task approach may be proposed for discriminating ADLs while learning the subject variability.

The performed comparison with respect to the one-dimensional CNN outlines the main advantage of the proposed methodology. The geometry of convolutional kernels restricts the captured range of dependencies between data samples, while the LSTM overcomes this limitation by exploiting their internal memories to capture long-range dependencies in variable-length input sequences.

The comparison with respect to traditional ML literature employed in [16] outlines the ability of the LSTM to automatically extract spatio-temporal information while reducing the time-consuming effort for preprocessing data and handcrafted features extraction. For instance, considering the same setting (i.e., Kyoto), the most improvement of the LSTM with respect to ML was achieved in Kyoto4, where a higher number of test days and activities were considered. This is in line with the advantage of DL approaches in terms of learning a representation of a huge amount of data with a large number of classes.

Future work could also test other similar datasets (e.g., [50]) that have already been used in literature for the HAR task [31].

In conclusion, given the evidenced, almost-equal HAR task performance, it would be advisable to prefer the LSTM-based model that offers fewer parameter complexities and computational costs.

References

References

[1] C. Röcker, M. Ziefle, A. Holzinger, Social inclusion in ambient assisted living environments: Home automation and convenience services for elderly user, in: International Conference on Artificial Intelligence, Vol. 1, 2011, pp. 55–99.

[2] World Health Organization, "World report on ageing and health", Geneva Switzerland, 2015.

[3] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, Z. Yu, Sensor-based activity recognition, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews). 42 (6) (2012) 790–808.

[4] L. Satpathy, A. Mathew, Technology to Aid Aging in Place – New Opportunities and Challenges, VDM Verlag, Saarbrücken, Germania, 2007.

[5] J. C. Augusto, H. Nakashima, H. Aghajan, Ambient intelligence and smart environments: A state of the art, Handbook of Ambient Intelligence and Smart Environments (2010) 3–31.

[6] D. J. Cook, J. C. Augusto, V. R. Jakkula, Ambient intelligence: Technologies, applications, and opportunities, Pervasive and Mobile Computing. 5 (4) (2009) 277–298.

[7] S. Ranasinghe, F. Al Machot, H. C. Mayr, A review on applications of activity recognition systems with regard to performance and evaluation, International Journal of Distributed Sensor Networks. 12 (8) (2016) 1550147716665520.

[8] S. Zolfaghari, R. Zall, M. R. Keyvanpour, Sonar: Smart ontology activity recognition framework to fulfill semantic web in smart homes, in: Second International Conference on Web Research, IEEE, 2016, pp. 139–144.

[9] D. Cook, K. D. Feuz, N. C. Krishnan, Transfer learning for activity recognition: A survey, Knowledge and Information Systems. 36 (3) (2013) 537–556.

[10] M.-C. Chang, N. Krahnstoever, S. Lim, T. Yu, Group level activity recognition in crowded environments across multiple cameras, in: Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance, IEEE, 2010, pp. 56–63.

[11] D. Di Paola, D. Naso, A. Milella, G. Cicirelli, A. Distante, Multi-sensor surveillance of indoor environments by an autonomous mobile robot, International Journal of Intelligent Systems Technologies and Applications 8 (1–4) (2009) 18–35.

[12] G. Okeyo, L. Chen, H. Wang, Combining ontological and temporal formalisms for composite activity modelling and recognition in smart homes, Future Generation Computer Systems. 39 (2014) 29–43.

[13] S. Ramasamy Ramamurthy, N. Roy, Recent trends in machine learning for human activity recognition—A survey, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. (2018) e1254.

[14] J. Yuen, A. Torralba, A data-driven approach for event prediction, Computer Vision. (2010) 707–720.

[15] J. Ye, G. Stevenson, S. Dobson, Kcar: A knowledge-driven approach for concurrent activity recognition, Pervasive and Mobile Computing 19 (2015) 47–70.

[16] D. J. Cook, Learning setting-generalized activity models for smart spaces, IEEE Intelligent Systems. 27 (1) (2012) 32–38.

[17] E. Chinellato, D. C. Hogg, A. G. Cohn, Feature space analysis for human activity recognition in smart environments, in: 12th International Conference on Intelligent Environments, IEEE, 2016, pp. 194–197.

[18] L. G. Fahad, S. F. Tahir, M. Rajarajan, Activity recognition in smart homes using clustering based classification, in: 22nd International Conference on Pattern Recognition, IEEE, 2014, pp. 1348–1353.

[19] D. J. Cook, N. C. Krishnan, P. Rashidi, Activity discovery and activity recognition: A new partnership, IEEE Transactions on Cybernetics. 43 (3) (2013) 820–828.

[20] P. Hu, J. Indulska, R. Robinson, An autonomic context management system for pervasive computing, in: Sixth Annual IEEE International Conference on Pervasive Computing and Communications, IEEE, 2008, pp. 213–223.

[21] J. Wen, Z. Wang, Sensor-based adaptive activity recognition with dynamically available sensors, Neurocomputing. 218 (2016) 307–317.

[22] L. G. Fahad, A. Khan, M. Rajarajan, Activity recognition in smart homes with self verification of assignments, Neurocomputing. 149 (2015) 1286–1298.

[23] I. Fatima, M. Fahim, Y.-K. Lee, S. Lee, Analysis and effects of smart home dataset characteristics for daily life activity recognition, Journal of Supercomputing. 66 (2) (2013) 760–780.

[24] G. Abebe, A. Cavallaro, Hierarchical modeling for first-person vision activity recognition, Neurocomputing. 267 (2017) 362–377.

[25] Y. Liu, L. Nie, L. Liu, D. S. Rosenblum, From action to activity: Sensor-based activity recognition, Neurocomputing. 181 (2016) 108–115.

[26] J. Modayil, T. Bai, H. Kautz, Improving the recognition of interleaved activities, in: 10th international conference on Ubiquitous Computing, ACM, 2008, pp. 40–43.

[27] Y. Zhang, Y. Zhang, E. Swears, N. Larios, Z. Wang, Q. Ji, Modeling temporal interactions with interval temporal bayesian networks for complex activity recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence. 35 (10) (2013) 2468–2483.

[28] S. Georgakopoulos, K. Kottari, K. Delibasis, V. Plagianakos, I. Maglogiannis, Pose recognition using convolutional neural networks on omni-directional images, Neurocomputing. 280 (2018) 23–31.

[29] J. Yu, C. Hong, Y. Rui, D. Tao, Multitask autoencoder model for recovering human poses, IEEE Transactions on Industrial Electronics 65 (6) (2018) 5060–5068.

[30] C. Hong, J. Yu, J. Wan, D. Tao, M. Wang, Multimodal deep autoencoder for human pose recovery, IEEE Transactions on Image Processing 24 (12) (2015) 5659–5670.

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

[32] R. Salakhutdinov, Learning deep generative models, Annual Review of Statistics and its Application 2 (2015) 361–385.

[33] M. M. Hassan, M. Z. Uddin, A. Mohamed, A. Almogren, A robust human activity recognition system using smartphone sensors and deep learning, Future Generation Computer Systems 81 (2018) 307–313.

[34] J. Wang, Y. Chen, S. Hao, X. Peng, L. Hu, Deep learning for sensor-based activity recognition: A survey, Pattern Recognition Letters.

[35] S. Ji, W. Xu, M. Yang, K. Yu, 3D convolutional neural networks for human action recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence. 35 (1) (2013) 221–231.

[36] S. Qian, H. Liu, C. Liu, S. Wu, H. San Wong, Adaptive activation functions in convolutional neural networks, Neurocomputing 272 (2018) 204–212.

[37] M. Yang, Y. Liu, Z. You, The euclidean embedding learning based on convolutional neural network for stereo matching, Neurocomputing. 267 (2017) 195–200.

[38] N. Y. Hammerla, S. Halloran, T. Ploetz, Deep, convolutional, and recurrent models for human activity recognition using wearables, arXiv preprint arXiv:1604.08880, (2016).

[39] A. Murad, J.-Y. Pyun, Deep recurrent neural networks for human activity recognition, Sensors. 17 (11) (2017) 25–56.

[40] J. K. Aggarwal, M. S. Ryoo, Human activity analysis: A review, Computing Surveys. 43 (3) (2011) 16.

[41] D. J. Cook, M. Schmitter-Edgecombe, Assessing the quality of activities in a smart environment, Methods of Information in Medicine. 48 (5) (2009) 480.

[42] L. Chen, C. D. Nugent, J. Biswas, J. Hoey, Activity recognition in Pervasive Intelligent Environments, Vol. 4, Springer Science & Business Media, 2011.

[43] I. Fatima, M. Fahim, Y.-K. Lee, S. Lee, A unified framework for activity recognition-based behavior analysis and action prediction in smart homes, Sensors. 13 (2) (2013) 2682–2699.

[44] K. Gayathri, K. Easwarakumar, S. Elias, Probabilistic ontology based activity recognition in smart homes using markov logic network, Knowledge-Based Systems. 121 (2017) 173–184.

[45] E. Nazerfard, D. J. Cook, CRAFFT: An activity prediction model based on bayesian networks, Journal of Ambient Intelligence and Humanized Computing. 6 (2) (2015) 193–205.

[46] C. Coppola, T. Krajník, T. Duckett, N. Bellotto, et al., Learning temporal context for activity recognition., in: 22nd European Conference on Artificial Intelligence, 2016, pp. 107–115.

[47] B. Minor, D. J. Cook, Forecasting occurrences of activities, Pervasive and Mobile Computing. 38 (2017) 77–91.

[48] H. Fang, C. Hu, Recognizing human activity in smart home using deep learning algorithm, in: 33rd Chinese Control Conference, IEEE, 2014, pp. 4716–4720.

[49] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, A. Baskurt, Sequential deep learning for human action recognition, in: International Workshop on Human Behavior Understanding, Springer, 2011, pp. 29–39.

[50] T. L. van Kasteren, G. Englebienne, B. J. Kröse, Human activity recognition from wireless sensor network data: Benchmark and software, in: Activity Recognition in Pervasive Intelligent Environments, Springer, 2011, pp. 165–186.

[51] M. Gallissot, J. Caelen, N. Bonnefond, B. Meillon, S. Pons, Using the multicom domus dataset, Ph.D. thesis, Laboratoire d'Informatique de Grenoble (2011).

[52] H. Alemdar, H. Ertan, O. D. Incel, C. Ersoy, ARAS human activity datasets in multiple homes with multiple residents, in: Seventh International Conference on Pervasive Computing Technologies for Healthcare, Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, 2013, pp. 232–235.

[53] A. Fleury, N. Noury, M. Vacher, Supervised classification of activities of daily living in health smart homes using svm, in: International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE, 2009, pp. 6099–6102.

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

[55] D. J. Cook, M. Schmitter-Edgecombe, CASAS Datasets, `http://ailab.wsu.edu/casas/datasets/`, accessed 06 September 2018.

[56] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation. 9 (8) (1997) 1735–1780.

[57] F. A. Gers, N. N. Schraudolph, J. Schmidhuber, Learning precise timing with lstm recurrent networks, Journal of Machine Learning Research. 3 (Aug) (2002) 115–143.

[58] G. Hinton, N. Srivastava, K. Swersky, Neural networks for machine learning, lecture 6a: Overview of mini-batch gradient descent, department of computer science university of toronto (2013).

[59] M. Schuster, K. K. Paliwal, Bidirectional recurrent neural networks, IEEE Transactions on Signal Processing. 45 (11) (1997) 2673–2681.

[60] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al., Google's neural machine translation system: Bridging the gap between human and machine translation, arXiv preprint arXiv:1609.08144, (2016).

[61] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, Journal of Machine Learning Research. 15 (1) (2014) 1929–1958.

Biography of the authors

Daniele Liciotti received the B.Sc. degree in Computer and Automation Engineering from Polytechnic University of Marche with a thesis entitled "Analisi di modelli dinamici per turbine eoliche" (supervisor Prof. Giuseppe Orlando).
In 2013, he received the M.Sc. degree in Computer and Automation Engineering from the Polytechnic University of Marche with a thesis entitled "Analisi automatica del comportamento dei consumatori in ambienti di retail intelligenti" (supervisor Prof. Emanuele Frontoni).
From November 2014 to October 2017 he was a Ph.D. Student at Polytechnic University of Marche, under the supervision of Prof. Emanuele Frontoni.

Michele Bernardini received the B.Sc. degree in Biomedical Engineering from the Polytechnic University of Marche with a thesis entitled "Ischemia miocardica acuta: dall' attivazione elettrica del cuore al segnale ECG" (supervisor Prof. Laura Burattini).
In 2016, he received the M.Sc. degree in Electronic Engineering from the Polytechnic University of Marche with a thesis entitled “Development of an automatic procedure to mechanically characterize soft tissue materials using visual sensors” in collaboration with Université Libre de Bruxelles (supervisors: Prof. Bernardo Innocenti, Prof. Emanuele Frontoni).
Since November 2017 he is Ph.D. Student at Polytechnic University of Marche (supervisor Prof. Emanuele Frontoni).

Luca Romeo received the Master of Science degree (cum laude) in Electronic Engineering at Università Politecnica delle Marche, in 2013, with a master thesis on the analysis of human-robot interaction (Ksera project).
From November 2014 to October 2017 he was a Ph.D. Student in Informatic Engineering at Università Politecnica delle Marche. From December 2017 he has a research fellowship at Università Politecnica delle Marche. He is also affiliated with the Department of CSML (Prof. Massimiliano Pontil) and C’MON (Prof. Cristina Becchio) from the Italian Institute of Technology (Genova).
His research interests focus on Machine learning, Affective Computing and Motion analysis.

Emanuele Frontoni is Professor of "Fondamenti di Informatica" and "Computer Vision” at the Università Politecnica delle Marche.
He received the Master degree in electronic engineering from the University of Ancona, Italy, in 2003. In the same year, he joined the Dept. of Ingegneria Informatica (DIIGA) at the Università Politecnica delle Marche, as a Ph.D. student in "Intelligent Artificial Systems". He obtained his Ph.D. in 2006 discussing a thesis on Vision-Based Robotics. His research focuses on applying computer science, artificial intelligence and computer vision techniques to mobile robots and innovative IT applications. He is a member of IEEE and AI*IA.

Daniele Liciotti

Michele Bernardini

Luca Romeo

Emanuele Frontoni