Representing Temporal Dependencies in Human Activity Recognition

Glenn Forbes[1], Stewart Massie[1], Susan Craw[1], Lucy Fraser[2], and Graeme Hamilton[2]

[1] School of Computing Science & Digital Media
Robert Gordon University, Aberdeen, UK
$\{g.r.forbes, s.massie, s.craw\}$@rgu.ac.uk
www.rgu.ac.uk/dmstaff/lastname-firstname
[2] Albyn Housing Society Ltd, Invergordon, UK

Abstract. Smart Homes offer the opportunity to perform continuous, long-term behavioural and vitals monitoring of residents, which may be employed to aid diagnosis and management of chronic conditions without placing additional strain on health services. A profile of the resident's behaviour can be produced from sensor data, and then compared over time. Activity Recognition is a primary challenge for profile generation, however many of the approaches adopted fail to take full advantage of the inherent temporal dependencies that exist in the activities taking place. Long Short Term Memory (LSTM) is a form of recurrent neural network that uses previously learned examples to inform classification decisions. In this paper we present a variety of approaches to human activity recognition using LSTMs and consider the temporal dependencies that exist in binary ambient sensor data in order to produce case-based representations. These LSTM approaches are compared to the performance of a selection of baseline classification algorithms on several real world datasets. In general, it was found that accuracy in LSTMs improved as additional temporal information was presented to the classifier.

Key words: Human Activity Recognition · Temporal Dependency · Smart Homes · Sensors · Time-series Data

1 Introduction

Smart Home technology is becoming increasingly popular but the focus to date has largely been on security and automation. However, there is real potential to employ smart home technology for health monitoring and management. Research has shown that there is a strong relationship between activities and behaviours that a person can undertake in their day-to-day lives and their future health and predicted lifespan [1]. The opportunity is to capture information on Smart Home residents by using sensors in smart homes to monitor a residents activities, e.g. room transitions, and behaviours, e.g. food preparation. A key advantage is that data is collected in the persons natural environment rather than in a more

artificial laboratory setting. Daily or weekly profiles of a resident's activities and behaviours can be captured over time, allowing trends in the data to be identified as a resident's profile changes; and a comparison of profiles to benchmark examples that indicate potential health concerns.

Sequences are important in activity recognition. An issue in making use of sequential sensor activation data is to develop effective approaches for harnessing temporal dependencies. The aim of this work is to identify and investigate the importance of different types of temporal relationships and propose specific representations and algorithms that can take advantage of these relationships.

A key focus of this work is on how to effectively take advantage of different forms of temporal relationships within the case representation. Several alternative approaches to classifying activities from low level, raw data inputs are investigated, with effective solutions identified which leverage temporal relationships. The key contributions of this work are:

- developing a novel temporal dependency-aware ML approach for activity recognition from event sequence sensor data;
- presenting a practical, flexible solution for building case-based representations for Human Activity Recognition.

The remainder of this paper is organised as follows. In Section 2 we review existing research on the use of Smart Homes for health monitoring. In Section 3 we consider methods for identifying activities from sensor data and in Section 4 we introduce six datasets that we use to evaluate alternative feature representations and learning algorithms on our activity recognition task. Finally, we draw our conclusions in Section 5.

2   Related Work

There has been a recent focus on deep learning in Human Activity Recognition. A key challenge for ML approaches is the acquisition of labelled data; it is expensive to manually annotate sensor data with activity labels, and variations in the labelling decisions of ambiguous sensor sequences may affect the overall quality of the dataset [2]. Smart home data presents its own challenges, as the data is often in the form an irregular sequence of events rather than a time sequence more typically of polled sensors with wearables. Only a few public datasets for activity recognition from Smart Home sensor data have been made available by MIT, CASAS and Van Kasteren [3–5].

As a form of Recurrent Neural Network (RNN), LSTMs offer the potential to capture the temporal relations encoded in the sequence of features and samples provided to the classifier [6]. In this paper the performance of a range of LSTM designs are investigated with various temporally-aware representations of sensor activation sequences.

The presence and importance of sequential events in the data from Smart Homes gives rise to opportunities to improve activity recognition by capturing the sequences as part of the representation. This may be explicitly by employing a

feature engineering approach, however, manually chosen features using heuristic processes may be overshadowed by modern deep learning research [6]. Ordóñez found that identifying the temporal dependencies that exist within human activity expressions was key to improving the performance of a HAR classifier. The experiments performed in their work made use of wearable sensor data, though similar relationships with no derived heuristics may exist between binary sensor activations [3]. In Bourobou's [7] work, HAR classification was performed on a simulated binary sensor dataset factoring in the temporal relation between subsequent, previous and overlapping activities. These temporal relations were identified separately and used to simulate data produced in real home environments. The relationships between temporally relevant activities were analysed using the calculated "importance degree" of each activity. While this approach yielded noteworthy results, additional research is still necessary to assess the efficacy of alternative approaches to manually engineering the influence of temporal dependence.

3 Identifying Activities from Sensor Data

Activities of Daily Living (ADLs) are regular behaviours which a person performs in day to day life. This can include getting out of bed, making a meal or grooming oneself. The specificity of an ADL can vary, with the classification of more complex ADLs (e.g. chopping vegetables) usually requiring additional sensor data. There is a trade-off between the number and cost of the sensor network and the specificity of ADLs that can be identified. On a wider scale, Human Activity Recognition (HAR) refers to the general task of identifying a person's activity expression from data, whereas ADLs usually refer to regular behaviours in a context around the home. HAR are typically discerned on a micro scale, such as standing up, sitting down or running; while ADLs refer to a more general form of activity. However, similar approaches can be applied to exploit patterns exhibited in data for both HAR and ADL classification.

The activity recognition task is typically broken down into two separate parts: windowing and ADL classification. Windowing involves separating the continuous sequence of sensor activations into smaller sequences likely to contain a single ADL expressions, while ADL classification is the process of assigning activity labels to each windowed sequences of sensor activations. We currently employ a rule-based approach for windowing and classification. As a first step the plan is to keep the windowing approach but change to a more flexible ML approach for classifying.

3.1 Rule-based Windowing

A popular approach to split data into windows is to use a sliding window [8]. This has been shown to be effective in real sensor datasets, however there is the potential for windows to overlap and not correctly represent realistic activity behaviours. Our approach to the windowing of long sequences of sensor activations

stems from the event-state system used in FITsense, which tracks the resident's movements and activity through the home. This is an additional filter layer between binary sensor activations and the classification task. Location labels associated with event-states can be used to split sequences of sensor activations as the resident transitions from one room to another. Due to the event-state system requiring states to depend on previous locations and behaviour, this rule-based classifier effectively makes use of the implicit temporal dependency in ADLs that occur sequentially. While this provides the benefit of reducing window overlap, the specificity of ADLs which can be captured is reduced.

3.2 ML

ML is now the standard approach employed for HAR. Advances in deep learning, recurrent and convolutional neural networks have directed the field for classification of complex sensor data, such as accelerometers and gyroscopes [9]. However, more basic sensor configurations, such as the binary sensor networks seen in smart home environments, have been well-served by Naive Bayes, Decision Trees and other established classification algorithms [3, 7].

While traditionally established ML methods can perform well in ADL classification tasks, they do not make use of the implicit temporal dependencies found in sensor data. The temporal relationships which exist in sensor data have relevance in ADL classification, as they offer an opportunity to extract additional useful knowledge from sensor activation data. Recurrent Neural Networking methods, specifically Long Short Term Memory (LSTMs), can make use of the temporal knowledge encoded in the sequences of sensor activations and ADLs which occur in training data.

In our data, we have identified 3 main types of temporal dependencies that are likely to be useful for activity classification:

- the order and sequence of sensor activations;
- the time of day at which events occur; and
- the order and sequence in which the ADLs take place.

We propose using a hybrid method to improve the temporal awareness of an ML-based ADL classifier. First, we plan to enrich sequential sensor data representations by adding relative timestamps between subsequent activations to representations for training data. Then by using LSTMs to learn intrabatch relationships between instances of ADLs.

4 Experiments

The aim of these experiments is to evaluate the performance of baseline classifiers on binary sensor datasets and compare their performance with LSTM implementations using implicit and explicit temporal knowledge. A selection of popular classifiers were used to establish the baseline performance of traditional

classifiers on this problem. These classifiers do not make use of the implicit temporal knowledge provided through the sequences in the data.

LSTMs can use previous learned examples to inform their decisions. We hypothesise that the performance of an ADL classifier can be improved by forming long term knowledge based on existing temporal dependencies. LSTMs were selected due to their demonstrated strength in time-series classification [10]. Additionally, LSTMs can make use of both the implicit and explicit temporal information in our data representations. Four LSTM configurations were compared to evaluate the performance impact of implicit and explicit temporal knowledge in ADL classification.

Iterations on our LSTM models were used to identify how performance could be improved by adding additional temporal information, both implicit and explicit. Implicit knowledge refers to the sequential information encoded in the order of sensor sequences in training data. The previous learned examples can influence the learning and prediction of future sequences in a traditional (or unidirectional) LSTM. Stateful LSTMs can make use of this additional implicit knowledge. Stateful LSTMs retain the hidden states of neurons between batches during training, allowing intrabatch dependencies to be inferred. In activity recognition data this allows the sequence of ADLs, as opposed to the sequence of sensor activations, to be encoded as additional knowledge. This may potentially be negatively affected by dataset gathering methods, such as non-contiguous collection. Significance in this model is placed on the order of ADLs; the resultant class of the classification task. Stateful LSTM implementations are marked in results with "S".

Explicit temporal knowledge in this context refers to the extension of the feature set to include a temporal data representation. The temporal data encoded in this feature could potentially represent the total length of a sensor sequence, the time of day at which it occurred, etc. For instance, by splitting the day into quadrants, a coarse timestamp identifying the quadrant of the day at which a sensor activation occurred could be an explicit representation. In initial experimentation, this specific feature was found to have little impact on the performance of our LSTM classifiers. Similar manually engineered features have had varying impact across multiple sensor sequence representations. In order to encourage the discovery of temporal dependencies, we implemented cumulative and relative timestamps into the representation. Manually engineering a feature relies on observations and assumptions of temporal importance in timestamp data, whereas fine timestamps allow for the algorithmic identification of key dependencies. Relative timestamps were determined to have the largest impact on the performance of our LSTM classifiers, with "E" marking experiments performed using representations containing explicit relative timestamps.

4.1 Labelled Datasets

These experiments make use of four publicly available datasets, alongside two datasets from the FITsense project. The datasets used in these experiments document instances of ADLs and their accompanying binary sensor activations

which have been captured using a variety of windowing methods. Details of the six datasets are shown in Table 1.

Table 1. Overview of the datasets used.

Dataset	Classes	Attributes	Instances
adlnormal	5	39	120
kasteren	7	14	242
tapia1	22	76	295
tapia2	24	70	208
fitsense1	7	13	744
fitsense2	7	13	990

CASAS[3] (adlnormal) This dataset contains the fewest classes, with 5 total activities observed by 39 independent sensors. While there is a small number of instances at 120, the activities tracked in the dataset are diverse enough to present little challenge for most baseline classification methods. As only 5 ADLs are tracked, large timegaps between activities can exist which may impact the performance of stateful LSTMs.

Van Kasteren[4] (kasteren) The kasteren dataset follows a structure most similar to that of fitsense1/2, with similar tracked activities and sensors. The ADLs expressed in this dataset have relevance to health monitoring and add a layer of complexity which may present a challenge in classification. Prepare_Breakfast and Prepare_Dinner are observably similar as activation sequences, however the time at which they are performed is important.

MIT[5] (tapia1/2) The most complex of the datasets used in these experiments due to the large number of sensors and classes. Several ADLs could be considered beyond the scope of capability for the sensor network (e.g. Going out shopping vs Going out for entertainment), however the additional complexity presents a useful challenge for classification.

FITsense/FitHomes[6] (fitsense1/2) These datasets contain the largest number of instances while using the least sensors of the datasets used in these experiments. The tracked activities were selected for health monitoring applications such as "sleeping", "grooming", and "foodprep".

---

[3] `http://casas.wsu.edu/datasets/adlnormal.zip`

[4] `https://sites.google.com/site/tim0306/kasterenDataset.zip`

[5] `http://courses.media.mit.edu/2004fall/mas622j/04.projects/home/thesis_data_txt.zip`

[6] `https://www.rgu.ac.uk/fitsense`

4.2 Experimental Design

WEKA was used to run a set of baseline classifiers on the datasets. Each dataset was converted from its original format to a zero-padded sequence of discrete values representing binary sensor activations in the ARFF format. Baseline algorithms were selected to evaluate the performance of a representative variety of classification techniques. LibSVM, J48, Bayes and k-NN were selected due to their established significance in the field. Each of these classifiers were run with default configurations as supplied by WEKA.

Temporally aware LSTM implementations were configured using Keras, using the Tensorflow backend and run on a system using an Nvidia GTX 1080 GPU. The LSTMs are trained using the "categorical_crossentropy" loss function and "adam" optimizer. Different batch size, units and epochs values were used for each dataset due to variation in sequence length, sensor makeup and activity composition. The batch sizes used ranged from 120-590, units ranged from 64-256, and the epochs to which each experiment was run ranged from 30-250. Stateful LSTMs were implemented using the "stateful" option in Keras while otherwise utilising the same configuration.

Baseline algorithms implemented in WEKA were run using 10-fold cross validation across the whole dataset. Metasequences of sensor activations can be broken up as the implicit temporal dependency between samples is not considered. Keras implementations were run using Leave One Out cross validation, with each dataset being split by day. This ensures folds contains contiguous sequences ensuring realistic meta-sequences are represented, with each fold starting and ending with "sleeping" ADLs. Each session of training was also repeated 3 times with fixed seeds to ensure repeatability.

4.3 Results

Table 2. Classification results (macro f1 scores).

Dataset	SVM	J48	Bayes	k-NN	LSTM	SLSTM	ELSTM	ESLSTM
adlnormal	0.898	0.934	**0.983**	0.910	0.932	0.951	**0.975**	0.918
kasteren	**0.901**	0.891	0.871	0.892	**0.874**	0.831	0.867	0.856
tapia1	0.162	**0.303**	0.246	0.248	0.212	0.202	**0.331**	0.287
tapia2	0.129	**0.314**	0.070	0.219	0.133	0.240	**0.359**	0.256
fitsense1	0.281	0.613	0.600	**0.667**	0.853	0.833	0.740	**0.864**
fitsense2	0.464	**0.620**	0.530	0.560	0.676	0.728	0.586	**0.752**

The results for the baseline classifiers in shown in Table 2. Each algorithm delivers the higest result on at least one dataset, however the overall winner is narrowly J48. This is due to its performance on more complex datasets such as tapia1/2 and fitsense2. Bayes and LibSVM demonstrate impressive performance on the adlnormal and kasteren datasets respectively. Performance of the baseline classifiers on the tapia datasets is relatively poor in comparison to the strong performance on others. Performance on the fitsense datasets is relatively good in comparison to tapia, however it still falls short of that seen on adlnormal and kasteren.

The LSTMs display a more balanced performance across all datasets. Slightly poorer performance can be observed between the top performing baseline classifiers and ELSTM implementations on adlnormal and kasteren datasets. On the more complex tapia datasets, improved performance over J48 can be observed in the ELSTM implementations. This improved performance on more complex datasets can also be seen in fitsense datasets, with ESLSTM being the clear winner over baseline classifiers. The overall winner in the LSTM implementations is ELSTM, with most results showing a leaning towards implementations utilising additional temporal knowledge.

4.4 Discussion

Of the baseline classifiers, J48 shows the most balanced performance across all datasets. Decision trees display strong performance in general activity recognition tasks. A potential future area of interest may be to investigate the methods by which temporal knowledge could be effectively represented in the training of decision trees for ADL classification.

The baseline classifiers did not have access to any temporal relationships, however on the simpler datasets (adlnormal and kasteren) they all achieved impressive results that outperformed LSTMs with some algorithms. This suggests that for simple classification tasks temporal relationship information is not required. However, on more complex tasks, including the FITsense data, the baseline algorithms' performance was poor highlighting the need to harness temporal relationships.

All LSTM implementations displayed more balanced performance than the baseline classifiers, with variants making use of additional temporal information giving better performance on the more complex classification tasks. Stateful LSTMs performed better than temporally unaware LSTMs across several datasets, however they did not win on any overall. While our temporally unaware LSTM narrowly won on the simpler kasteren dataset, ELSTM gave the best overall performance on the 4 publicly available datasets. This highlights the importance of capturing specific event timings as part of the representation for more complex tasks.

The key motivation for this work is to achieve good performance on FITsense data. On fitsense 1/2, ESLSTM was a clear winner, highlighting the importance of the activity sequences for this data. This is beacause the FITsense datasets

are different to the others in that they are a formed from a continuous time-stream and have contiguous windows. The inclusion of null or "none" ADL states ensures the complete sequence of activities is retained. As a result, the implicit meta-sequences of ADLs which occur in the data can be effectively used as an additional source of temporal knowledge.

In conclusion, the inclusion of relative time-stamps as explicit temporal information improved performance in most scenarios. This approach to temporal knowledge representation appears to have been successful in encouraging the discovery of temporal relations. The combination of implicit and explicit temporal representation performs best on the fitsense datasets, which are completely contiguous.

5   Conclusion

A key focus of the work is to develop case-based representations from simple sensor network inputs that can effectively capture temporal relationships in order to support improved ADL classifications. Specifically, we have presented LSTM solutions for providing additional implicit and explicit temporal knowledge to an ADL classifier, and compared their performance to established baseline algorithms. The proposed methods were evaluated on publicly available datasets, alongside our own FITsense datasets labelled using a hyrbid of rule-based windowing and manual sequence annotation.

The 3 key insights found in this work are:

- Additional temporal information has a positive impact on the performance of ADL classifiers, evidenced by ESLSTMs demonstrating the highest performance of any classifier used in our experiments.
- The method by which data is collected has a strong impact on the performance of ADL classifiers. Temporally aware classifiers perform best on contiguous datasets which capture uninterrupted sequences of activities, such as fitsense1/2.
- Allowing ADL classifiers to infer temporal dependencies results in better performance rather than manually engineering temporal features based on assumptions.

Future work could potentially investigate the production of lower level representations for deep learning to make use of additional unidentified temporal dependencies, and using other deep learning classifiers such as Convolutional Neural Networks.

References

1. N. K. Suryadevara and S. C. Mukhopadhyay. Wireless sensor network based home monitoring system for wellness determination of elderly. *IEEE Sensors Journal*, 12(6):1965–1972, June 2012.

2. Randy Joseph Rockinson. Activity Recognition with End-User Sensor Installation in the Home. Master's thesis, Massachusetts Institute of Technology, 2008.

3. Emmanuel Munguia Tapia, Stephen S. Intille, and Kent Larson. Activity recognition in the home using simple and ubiquitous sensors. In Alois Ferscha and Friedemann Mattern, editors, *Pervasive Computing*, pages 158–175, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

4. Diane Cook, M Schmitter-Edgecombe, Aaron Crandall, Chad Sanders, and Brian Thomas. 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*, pages 1–7, 2009.

5. Tim Van Kasteren, Athanasios Noulas, Gwenn Englebienne, and Ben Kröse. Accurate activity recognition in a home setting. In *Proceedings of the 10th international conference on Ubiquitous computing*, pages 1–9. ACM, 2008.

6. Francisco Javier Ordóñez and Daniel Roggen. Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors (Switzerland)*, 16(1), 2016.

7. Serge Thomas Mickala Bourobou and Younghwan Yoo. User activity recognition in smart homes using pattern clustering applied to temporal ANN algorithm. *Sensors (Switzerland)*, 15(5):11953–11971, 2015.

8. Jessamyn Dahmen, Brian L. Thomas, Diane J. Cook, and Xiaobo Wang. Activity learning as a foundation for security monitoring in smart homes. *Sensors (Switzerland)*, 17(4):1–17, 2017.

9. Ming Zeng, Le T. Nguyen, Bo Yu, Ole J. Mengshoel, Jiang Zhu, Pang Wu, and Joy Zhang. Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors. *Proceedings of the 6th International Conference on Mobile Computing, Applications and Services*, 6, 2014.

10. Felix A Gers, Douglas Eck, and Jürgen Schmidhuber. Applying lstm to time series predictable through time-window approaches. In *Neural Nets WIRN Vietri-01*, pages 193–200. Springer, 2002.