Neural Computing and Applications
https://doi.org/10.1007/s00521-020-04737-6

S.I. : ENGINEERING APPLICATIONS OF NEURAL NETWORKS 2018

Tracking changes in user activity from unlabelled smart home sensor data using unsupervised learning methods

Prankit Gupta[1] · Richard McClatchey[1] · Praminda Caleb-Solly[1]

Received: 22 January 2019 / Accepted: 10 January 2020

Abstract

This paper investigates the utility of unsupervised machine learning and data visualisation for tracking changes in user activity over time. This is done through analysing unlabelled data generated from passive and ambient smart home sensors, such as motion sensors, which are considered less intrusive than video cameras or wearables. The challenge in using unlabelled passive and ambient sensors data for activity recognition is to find practical methods that can provide meaningful information to support timely interventions based on changing user needs, without the overhead of having to label the data over long periods of time. The paper addresses this challenge to discover patterns in unlabelled sensor data using kernel density estimation (KDE) for pre-processing the data, together with t-distributed stochastic neighbour embedding and uniform manifold approximation and projection for visualising changes. The methodology is developed and tested on the Aruba CASAS smart home dataset and focusses on discovering and tracking changes in kitchen-based activities. The traditional approach of using sliding windows to segment the data requires a priori knowledge of the temporal characteristics of activities being identified. In this paper, we show how an adaptive approach for segmentation, KDE, is a suitable alternative for identifying temporal clusters of sensor events from unlabelled data that can represent an activity. The ability to visualise different recurring patterns of activity and changes to these over time is illustrated by mapping the data for separate days of the week. The paper then demonstrates how this can be used to track patterns over longer timeframes which could be used to help highlight differences in the user's day-to-day behaviour. By presenting the data in a format that can be visually reviewed for temporal changes in activity over varying periods of time from unlabelled sensor data, opens up the opportunity for carers to then initiate further enquiry if variations to previous patterns are noted. This is seen as an accessible first step to enable carers to initiate informed discussions with the service user to understand what may be causing these changes and suggest appropriate interventions if the change is found to be detrimental to their well-being.

Keywords Human activity recognition · Unlabelled sensor data · Data visualisation · Unsupervised learning

1 Introduction

With a growing shortage of carers and an ageing population, there is an urgent need to explore how smart sensing technologies could be utilised to support and maintain a high quality of agile and responsive care. Accordingly, researchers have been developing ambient assisted living (AAL) technology which utilises data from a range of smart home (SH) sensors to support people with long-term conditions to live independently [1]. The kitchen is usually the centre of user activity, particularly for those who are still managing to live independently. Additionally, most frequently occurring household injuries for vulnerable people occur in the kitchen, which can lead to loss of confidence in performing kitchen activities over time and moving to a nursing home [2, 3]. As such, tracking activities in the kitchen over time can provide the requisite baseline data for identifying early indicators of changes which might require interventions. Early intervention can

✉ Prankit Gupta
prankit.gupta@brl.ac.uk

[1] Faculty of Environment and Technology, University of the West of England, Coldharbour Ln Stoke Gifford, Bristol BS16 1QY, UK

Published online: 25 January 2020

prevent, and pre-empt, more serious issues from happening in the future.

A large area of AAL research is focussed on performing human activity recognition (HAR) from SH sensor data. This includes detecting activities offline, after they are finished, as well as detecting activities in real time as they occur. Real-time HAR is essential for interventions such as assistive prompts, while offline HAR is useful for tracking changes in user behaviour over time, detecting abnormal behaviour, as well as performing wellness evaluations.

The SH sensors used for HAR can be broadly categorised into wireless sensor networks (WSNs), body sensor networks (BSNs) and video-based solutions. WSNs comprise sensors that are integrated into the environment of the user such as passive infrared (PIR) motion sensors, magnetic contact sensors, and temperature sensors. Generally, a large number of WSNs are required to be present in order to perform HAR [4]. BSNs comprise sensors which can be present on the user, such as wearables which can provide accelerometer and GPS data along with the users' physiological information. Although the data provided by BSNs can be crucial for performing HAR, end-users can often forget to wear the sensors or charge them, or consider them intrusive. Video-based solutions provide the most context on the user and can range from RGB-D data to thermal imaging; however, they are generally considered an invasion of privacy by end-users [5, 6]. As cost-effective WSNs are becoming more commonly available as consumer products and are considered more acceptable than video-based solutions, exploring and developing their utility as part of an effective AAL technology solution to support users for living independently is a crucial next step.

There is a variety of existing research into HAR which has utilised supervised learning techniques using WSNs, BSNs, as well as video-based solutions with promising results. The problem with supervised learning is that it requires large amounts of user-annotated or labelled sensor data for training. This is often difficult to obtain for each individual user the system needs to be deployed for, and the subsequent trained classifier is also unable to adapt to changes in user behaviour without re-training with more labelled data. A common approach when collecting data for training classifiers requires the user to self-report or log activities through a diary, which is then used to annotate the data [7]. This introduces issues related to the reliability of the labels, as the user may forget to label every activity he/she performs or may not provide sufficient detail describing the activity [7]. This is evident in many user-annotated public smart home datasets where a simple "meal preparation" label is provided that can encompass a range of different types of cooking activities. Lastly, self-reporting of activities can be a tiring and tedious task, particularly when required to be conducted over many weeks or months and may not be possible for end-users with cognitive impairments. As such, researchers in this field are also investigating the use of unsupervised learning techniques, with a view to eliminating the need for labelling SH data. However, most of the existing research studies that have shown promising results used context-rich information obtained from BSNs and video-based solutions, and not WSNs. Researchers such as Fiorini et al. [8] used unsupervised learning with WSNs; however, in this case the authors were looking for an overall user "busyness" metric rather than individual user activity patterns.

This paper presents a novel approach for analysing unlabelled smart home sensor data, focussed on discovering patterns in user activity by analysing each of the days of the week separately over three 12-week periods. The approach presented in this paper is developed and tested on a total of 203 days of data from the kitchen-based sensors in the Aruba CASAS dataset [9]. By disregarding any labels present in the dataset for the visualisation, we seek to identify and understand sub-patterns that might exist with a view to interpreting user activity over time. The scope of this paper is to be able to inform the process of inquiry by the care provider for early intervention if variations to previous patterns are noted. This is seen as an accessible first step to enable carers to initiate informed discussions with the service user to understand what may be causing these changes and suggest appropriate interventions if the change is detrimental to their well-being.

The rest of the paper is structured as follows, Sect. 2 reviews existing HAR and data mining techniques in more detail; Sect. 3 provides a description of the Aruba CASAS dataset as used in the study; Sect. 4 describes the methodology, the data pre-processing and feature selection, and data visualisation techniques for discovering user activity patterns; Sect. 5 presents the results and discussion; and finally Sect. 6 summarises the conclusions and discusses future work.

2 Background and prior work

This section reviews HAR techniques in more detail, while also reviewing data mining techniques which have been used for applications other than HAR, utilising unsupervised learning.

In order to perform HAR, periods of sensor data events that may represent activities must be extracted first. Traditional approaches for this include the use of sliding time and sensor windows as used by Yala et al., Cook and Krishnan [7, 8]. These sliding windows are generally used for training supervised learning systems when activity labels are present, as the sliding windows can be chosen based on the activity labels present in the data, and windows containing noise can be removed manually. However, due to this, they are not as well suited for unlabelled data as it would be difficult to identify windows that contain noise. The lengths of these windows are also often fixed which makes the activity recognition system highly sensitive to variance in the distribution of sensor events throughout the day. Therefore, it is important to investigate alternative approaches for extracting periods of sensor data events of variable lengths.

An alternative approach is presented by Soulas et al. [6] for discovering "episodes" of user activities along with their periodicity and variability. The authors use an episode length of 30 min which essentially acts as a time window for extracting sensor data which may belong to an episode. However, this is left as a parameter to be set by the user depending on their daily habits. Along with this, Soulas et al. also define five additional parameters which need to be set by the user and the user's physician in order for the algorithm to work. The authors acknowledge that setting unsuitable parameters can lead to missing interesting information and other automated candidate episode generation techniques need to be investigated. Nevertheless, the paper highlights the need for HAR algorithms that do not require priori knowledge on the user. They also provide an analysis into the variability and repeatability of user behaviour present in the public SH datasets; however, their approach for this requires considerable hand-tuning of the learning methods.

In the work presented by Gupta and Caleb-Solly [10], sensor data was analysed by room only, and treated as 1D time series data per room, only comprising of sensor event timestamps. An alternative approach to sliding windows in this case would be to find and extract periods of high-density present in the sensor data which could potentially represent activities. As the sensor data can be treated as 1D time series, kernel density estimation (KDE), as first proposed by Rosenblatt, can be a powerful tool for extracting periods of sensor events which can potentially represent an activity [11]. KDE is a nonparametric method for estimating the probability density function of a random variable, and as such can be used to detect time periods of high-density present in 1D data. This overcomes the issue of deciding the size of sliding windows and has the added benefit of identifying only high periods of sensor activity and disregarding the rest as noise. KDE has two parameters—kernel function and the bandwidth. The kernel function must be chosen based on the properties of the data, while the bandwidth can be selected using Silverman's rule [12]. As these parameters can be derived statistically, KDE can be a potential alternative to traditional fixed-size sliding windows for extracting sensor data.

Once periods of sensor data are extracted, the next step is visualising the data. In recent years, new visualisation techniques have been introduced which have superseded existing techniques such as Self-Organising Maps (SOM's) and principal component analysis (PCA) in certain applications. These visualisation techniques include t-distributed stochastic neighbour embedding (t-SNE) [13] and uniform manifold approximation and projection (UMAP) [14], both of which are nonlinear dimensionality reduction techniques. T-SNE has often been the primary choice for researchers for visualising high-dimensional data in 2D and is noted for preserving the local structure of the data. UMAP on the other hand is a much newer technique and is capable of preserving both local and global structure of the data [15]. These techniques are particularly relevant when dealing with unlabelled data, as they can help to discover whether there are any meaningful features and potential clusters present. However, it must be noted that even though both t-SNE and UMAP are both useful choices for visualisation, clustering based on their output is generally not recommended, as density information is often lost during the process [16]. A useful technique is also presented by Fiorini et al. [8], where radar graphs were constructed from motion sensor data which can be used to facilitate a quick visual review of the sensor data. This technique can be used in conjunction with other visualisation techniques such as UMAP, to gain further insight into the sensor data.

To summarise, in this section the potential benefits of KDEs to replace sliding windows for extracting sensor data and the use of t-SNE/UMAP for visualising unlabelled data are highlighted.

The next section provides a description of the Aruba CASAS smart home dataset, which will be used for developing, as well as testing the unsupervised learning methodology presented in this paper.

3 Selection and description of the public smart home dataset

This section provides details of the selection process for the smart home dataset, and activities which were selected for use in this study. For this research, we focussed on the Washington State University's Centre for Advanced Studies in Adaptive Systems (CASAS) [9] dataset collection. This collection comprises a range of labelled, partly labelled, or unlabelled activity data, collected over varying time periods. The activities in these datasets are scripted or unscripted. The work presented in this paper is focussed on unscripted "daily life" datasets. Additionally, datasets which use BSNs or video cameras were not considered as the focus in this research is on utilising less intrusive WSNs, such as PIR motion sensors. Five public datasets met these criteria.

A further search was conducted for these five CASAS datasets on IEEE[1] with keywords ((((Smarthome) OR smart-home) AND '*nameofdataset*') AND CASAS). This revealed the Aruba dataset as the most frequently used dataset with 31 search results, and Milan as the second most frequently used with 15 search results. Based on this, both Milan and Aruba were shortlisted. The Aruba dataset has a total of 220 days of continuous data, while the Milan dataset has a total of 72 days of data. However, Milan is missing 11 days of data, while the Aruba dataset has continuous data with no missing days. The missing days could affect the performance of the HAR algorithm as it is crucial to analyse consecutive days in order to pick up repeating activity patterns. The Aruba dataset was therefore selected for developing and testing the unsupervised learning techniques presented in this paper.

The Aruba dataset consists of data from a total of thirty-nine sensors, out of which thirty-four are PIR sensors and five are temperature sensors. In this paper, only the PIR sensor data, which represent the occupant's physical movement in the vicinity of the sensor, is analysed. Therefore, after excluding temperature sensor data over the period of 220 days, a total of 1,602,980 sensor events are present out of which 849,579 sensor events ($\approx$ 53%) are not annotated with any activity labels in the dataset. Previous studies have often discarded these unlabelled sensor events when performing HAR as activities detected using the unlabelled data cannot be verified [17].

There are a total of 11 activity labels present in the Aruba dataset (Fig. 1). The primary kitchen activity labels are "Meal_Preparation" and "Wash_Dishes. There are a total of 1606 instances of the "Meal_Preparation" activity and only 65 instances of the "Wash_Dishes" activity. In previous studies, this imbalance has caused classifiers to misclassify the "Wash_Dishes" as "Meal_Preparation" activity [10].

As this study is focused on kitchen activities, only kitchen sensor data was analysed, which includes "Meal_Preparation" and "Wash_Dishes" activity labels. These event data represented by these two labels was further analysed to verify which sensors were associated with these labels in the Aruba dataset. Both "Meal_Preparation" and "Wash_Dishes" labels were primarily based on only kitchen sensors (five PIR sensors) being triggered over the entirety of the dataset; all other sensors in the house were associated with less than 5% of both the activity labels. This supports the approach previously presented by Gupta and Caleb-Solly [10], in which only kitchen sensor data was analysed when performing HAR for kitchen activities, considerably reducing the noise and amount of the data required to be processed. It should be noted that no unlabelled sensor data was removed by hand, as it was left to the unsupervised machine learning techniques to identify noise. As stated previously, this is different to previous studies by other researchers using this dataset, who removed all unlabelled data from the analysis as the activity represented by that data could not be verified [17]. The approach of retaining unlabelled data better reflects a real-world scenario, where a dataset is likely to contain unlabelled instances.

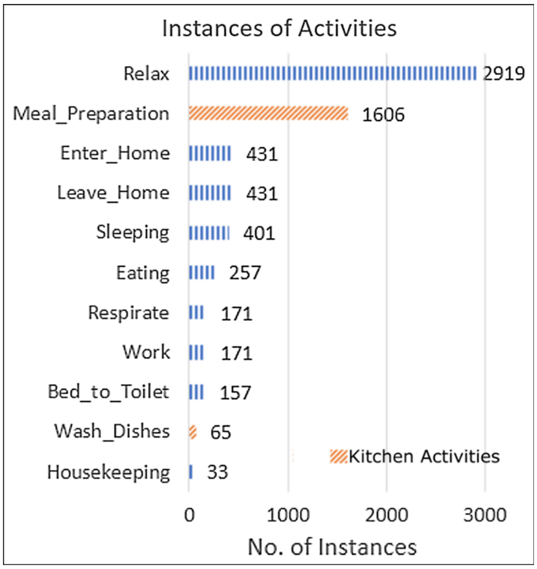


Fig. 1 Total instances of activity labels in the Aruba dataset

4 Methodology

This section outlines the methodology followed in this paper which includes extracting temporal clusters using kernel density estimation from sensor data, feature selection, and the use of data visualisation techniques.

All the algorithms were written in Python using various machine learning libraries which are referenced throughout the paper.

4.1 Extracting temporal clusters of sensor events using KDE

Over the past decade, as research into AAL and SHs has grown, various new concepts and terminology have been introduced to the field. In this paper, some of these existing concepts have been further developed, such as that of a temporal cluster. This study presents a method for extracting periods of high-density present in the temporal sensor data, which have been defined as temporal clusters (Fig. 2). Therefore, a temporal cluster ($\boldsymbol{TC_i}$) is a set of

[1] https://ieeexplore.ieee.org/.

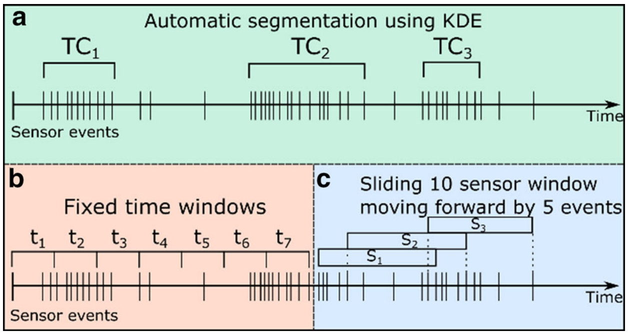


Fig. 2 **a** Sensor segmentation using KDE to extract temporal clusters (TC_i), **b** fixed time windows (t_i), and **c** sliding sensor windows with a length of 10 sensor events and sliding forward by 5 sensor events (s_i)

sensor events occurring close together $\{s_1, s_2, s_3, \ldots s_n\}$ that could potentially represent an activity.

In this study, temporal clusters are used with a view to identify activity patterns which might not have been represented by the user labels, but might still represent specific user activities or behaviour.

For developing this temporal cluster extraction approach using KDE, the Aruba dataset was divided into a training and test set. Days 10 to 42 were used as the training set and days 53 to 81 were used as the test set.

KDE requires the selection of a kernel and the kernel's bandwidth. After analysing the training set, the Epanechnikov kernel [18] was empirically selected for the algorithm. The Silverman's rule [12] for automatically selecting the bandwidth was also empirically adjusted to:

$$bw = 0.07\hat{\sigma}n^{-1/5}$$

where $\hat{\sigma}$ is the standard deviation of the sample, n is the sample size and bw is the bandwidth. The KDE temporal cluster extraction technique is illustrated in Fig. 3. This figure shows an example of KDE temporal cluster extraction process for the morning hours of 8 am to 10 am for a selected day from the Aruba dataset. For the experiment, KDE was used to generate a density curve for the whole day which was then used to extract temporal clusters as shown. Following this, all the sensor events included within the mid-height of the peaks were extracted as a single temporal cluster (Fig. 3d). The mid-height of the peaks were calculated as 50% of the height of the peak relative to whichever comes first—the last local minima before a local maxima higher than the current peak, or the global minima. This ensures that a peak which is higher than other peaks that follow it, extracts a larger temporal cluster as is the case for peak 2 in Fig. 3d. Mid-heights that contained less than two sensor events or lasted less than 60 s were discarded as noise. This 60 s threshold value for noise along with the mid-height of the peak was selected after analysing the training data with different values and peak heights until all the "Meal_Preparation" activities could be identified.

A feature of using KDE is that it can also discover temporal clusters which may represent interleaved and overlapping activities. An example of this can be seen in Fig. 3d where temporal cluster from peak 3 overlaps a larger temporal cluster from peak 2. The *stats* module from the *SciPy*[2] was used to perform KDE in Python [19].

4.2 Feature selection

The next step was to select features from the temporal clusters extracted as described in the previous section. These features are listed in Table 1.

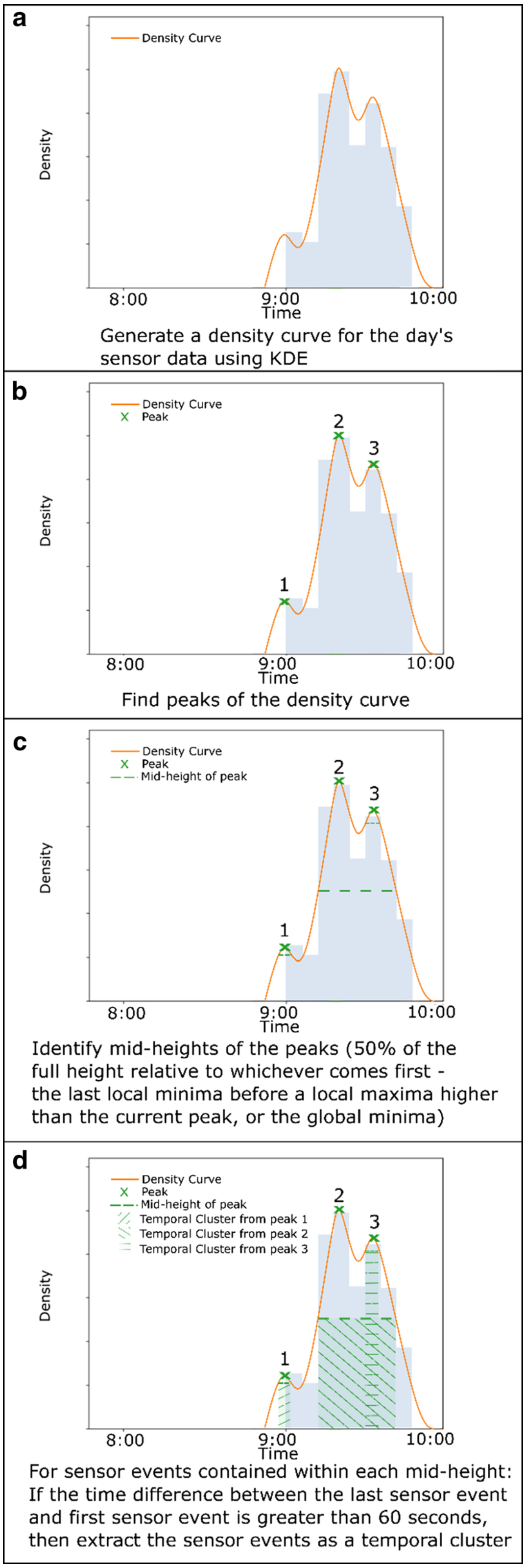
This feature set consists of eight features, the first three being—duration (length) of the temporal cluster, the variance in each temporal cluster based on timestamps of the sensor events, and start time of the temporal cluster. The start time was corrected to the hour closest to the first timestamp of the temporal cluster. The last five features were total number of events from each sensor separately in the Kitchen. All features were normalised between 0 and 1.

4.3 Visualisation using UMAP

In order to visualise the behavioural changes by day-of-the-week, UMAP was performed to generate data points in a two-dimensional space from the eight-dimensional feature sets of the temporal clusters. It was hypothesised that using a day-of-the-week level of granularity might help to better track changes in the longer term, because as shown in Fiorini et al. [8], there can be marked differences between weekday and weekend routines.

A 12-week (3 month) period was considered which ensured that there were enough data points for identifying repeating patterns for each day of the week. Four such periods of 12 weeks were then compared to verify whether the activity patterns persist and whether any slight changes were apparent. This means analysing four sets of 12 Mondays, 12 Tuesdays, and so on. The first three of these periods were overlapping and moving forward by 1 week at a time as follows—weeks 2 to 14, 3 to 15, and 4 to 16. This was done with a view to analyse small shifts in the user's daily routine. The last period did not overlap with the first three periods and consisted of weeks 17 to 29. This was done to determine whether, if at all, user behaviour may have changed after a longer non-overlapping time period.

[2] https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.gaussian_kde.html.



Generate a density curve for the day's sensor data using KDE

Find peaks of the density curve

Identify mid-heights of the peaks (50% of the full height relative to whichever comes first - the last local minima before a local maxima higher than the current peak, or the global minima)

For sensor events contained within each mid-height: If the time difference between the last sensor event and first sensor event is greater than 60 seconds, then extract the sensor events as a temporal cluster

Fig. 3 Three temporal clusters extracted from the density curve generated using KDE for a single morning, overlaid with a histogram of sensor event timestamps

Table 1 Features selected from each temporal cluster

No.	Feature
1	Duration of temporal cluster
2	Variance of temporal cluster
3	Start time of temporal cluster (hour)
4	Total sensor events for kitchen sensor 1
5	Total sensor events for kitchen sensor 2
6	Total sensor events for kitchen sensor 3
7	Total sensor events for kitchen sensor 4
8	Total sensor events for kitchen sensor 5

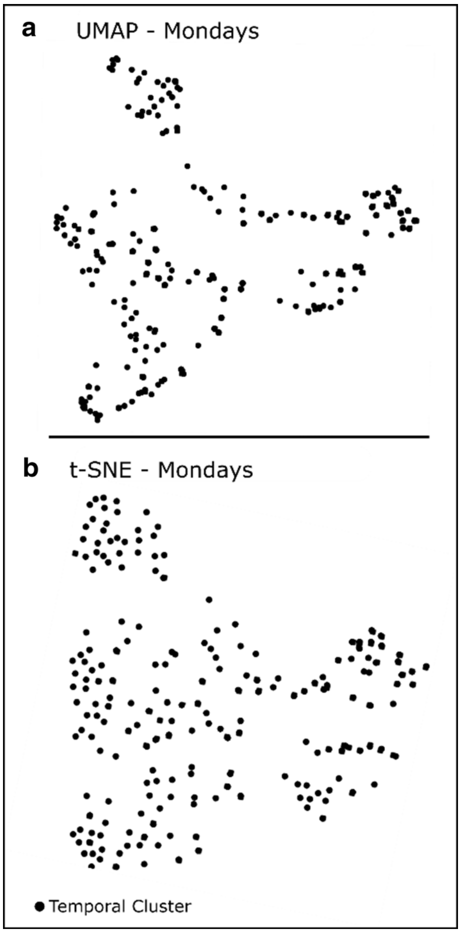


Fig. 4 **a** Top—UMAP, **b** Bottom—t-SNE (both projections are for weeks 4 to 16)

Figure 4a shows an example of Mondays for weeks 4 to 16 for UMAP. The parameter “n_neighbours” was set to 15 and “min_dist” was set to 0.1 empirically.

T-SNE (Fig. 4b) was also performed for comparison using the same data set to verify that the UMAP plot does not contain spurious artefacts. The perplexity parameter of t-SNE was determined empirically and set to 25.

It can be seen in Fig. 4 that the plots created by UMAP and t-SNE are visually similar. T-SNE plots also appeared to have a less visually discernible morphology as can be seen in Fig. 4b, which makes them harder to interpret for the sensor data. It must also be noted that t-SNE is very sensitive to the perplexity parameter and as such makes it difficult to obtain consistent and reliable results [16]. For these reasons, UMAP was favoured for visualising the patterns of activities clusters.

5 Results and discussion

This section presents the results of the KDE for extracting temporal clusters, as well as the UMAP visualisations.

5.1 KDE: extracting temporal clusters

This section presents the results of using KDE temporal cluster extraction, as performed on the Aruba test dataset for each day individually. This technique was tested on a consecutive 28-day period. Using the KDE temporal cluster extraction technique, a total of 454 temporal clusters were extracted for this period (Table 2). These temporal clusters included 100% of the labelled activities present, within an error of + or − 5 min as compared to the timestamps of the activities in the dataset. 211 additional temporal clusters (46% of the total) were also discovered, which were not associated with an activity label.

Researchers in the past have either labelled the unlabelled periods of sensor activity as an additional “Other” activity or have removed them completely [17]. In such studies, the accuracy of the activity recognition system is significantly impacted due to the presence of noise in the unlabelled sliding windows being classified. In the study presented in this paper, temporal clusters that contained unlabelled data were not removed but were included in the analysis as they could potentially represent activities that are not labelled, yet are of significance in representing the user’s behaviour. Periods of low sensor activity in the Kitchen, which can be viewed as noise and not pertaining to any important activity information, were automatically removed by this technique as the density was too low to generate a temporal cluster (as explained in Sect. 4.1).

The next subsection presents the results of UMAP.

Table 2 KDE results for the test period: days 53 to 81 (total 28 days)

Labelled activities in the test period	243
Temporal clusters extracted (labelled)	243
Temporal clusters extracted (unlabelled)	211
Total temporal clusters extracted	454

5.2 UMAP visualisation

This section presents the results of UMAP visualisation. Figure 5 shows UMAP plots generated for each day of the week, for four 12-week periods (Period 1 (P1): weeks 2 to 14, Period 2 (P2): 3 to 15, Period 3 (P3): 4 to 16 and Period 4 (P4): weeks 17 to 29).

Each data point in the plot represents a temporal cluster which was extracted through KDE. As UMAP is primarily used for visualisation and clustering is generally not recommended [16], the analysis included for this approach is therefore based only on what is visually discernible.

As can be seen from Fig. 5, the UMAP plot for each day of the week has a slight triangular morphology (most evident in Tuesdays). However, each day of the week still has a distinct visual morphology that persists for at least the first three overlapping 12-week periods. Additionally, it can be observed that plots for Period 4 (weeks 17 to 29) in Fig. 5 are visually different compared to the plots for the preceding three periods, with the exception of Fridays. While we can’t conclusively determine the cause of this difference, noting of the presence of similarities and differences by the carer could be used as a mechanism to prompt further investigation through a discussion with the service user. For Mondays, Fig. 6 shows the UMAP changing over time from week 2 to 29. Each plot comprises data from a 12-week period, with a step-size of 3 weeks. There is a gradual, but visually discernible shift in the UMAP pattern over time.

When comparing the number of temporal clusters between the individual days of the weeks over all the four time periods, it can be seen in Fig. 7 that the number of temporal clusters is lower on Wednesdays and Thursdays. As the number of temporal clusters is indicative of the overall level of activity, this information could provide useful insight for the carer as to user’s different activity levels over the weeks.

Furthermore, in Fig. 7 a trend of a reduced number of temporal clusters for Period 4 can be noted when compared to the previous periods 1, 2 and 3. This is particularly evident for Tuesdays and Fridays.

5.2.1 Radar graph comparison

In Fig. 5, it can be seen that in addition to the overall UMAP cluster morphology, the level of dispersion of the points is also different for different days of the week. For example, when comparing Mondays to Thursdays in Fig. 5, a difference in the dispersion of points between Mondays and Thursday is visually discernible, i.e., the UMAP for Mondays has areas of varying density of points, while the UMAP for Thursdays is comprised of more

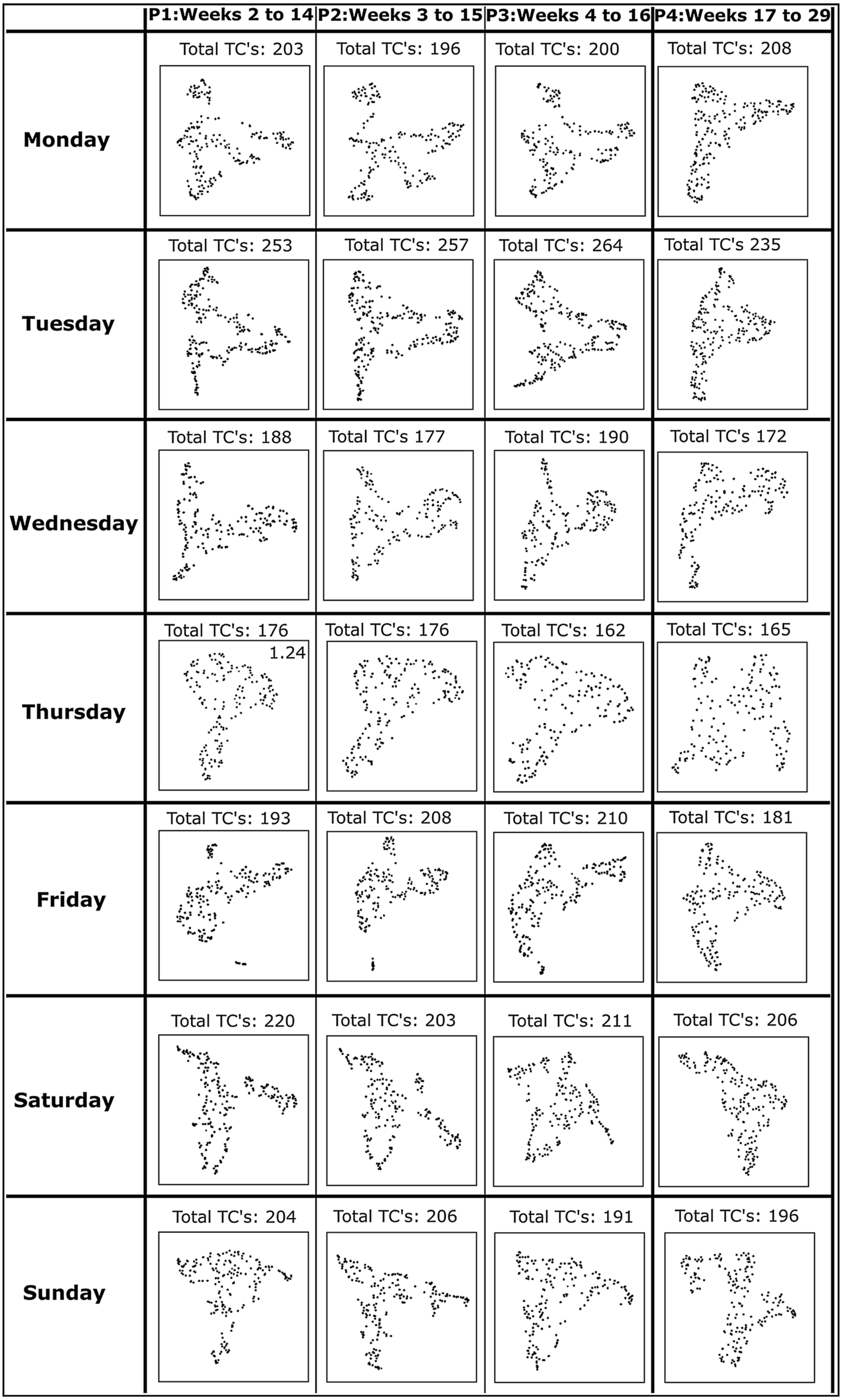


Fig. 5 UMAP visualisations for each weekday for three overlapping 12-week periods, and one separate 12-week period. Total TC's—total number of temporal clusters

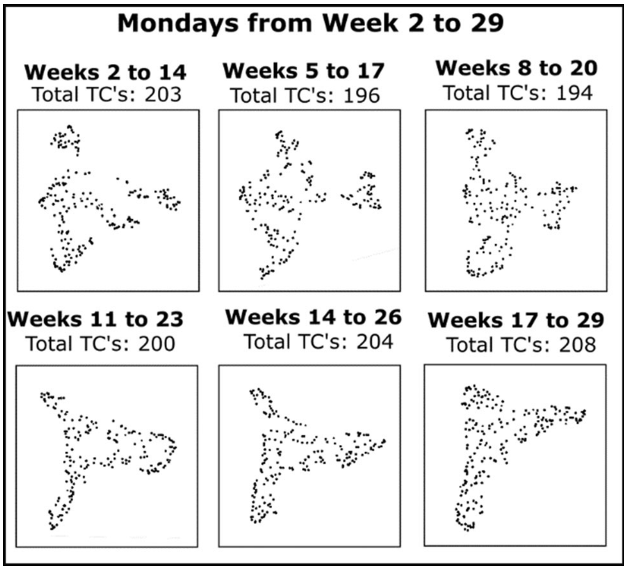


Fig. 6 UMAP for Mondays for 12 week periods moving forward by 3 weeks at a time, from week 2 to 29

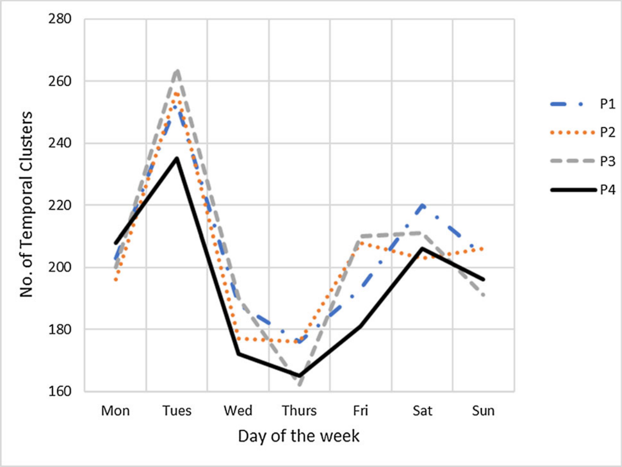


Fig. 7 Graph comparing the number of temporal clusters between the four time periods (P1, P2, P3 and P4)

uniformly distributed points. This could be partially explained by the lower number of temporal clusters present on Thursdays as shown in Fig. 7; however, Thursday for P4 has more temporal clusters than P3, but the data points in the former are still more dispersed, with a less distinct morphology. To gain further insight into these differences, radar graphs were generated for Mondays and Thursdays to identify the total number of temporal clusters at different times of day (ToD), similar to the approach presented by Fiorini et al. [8].

The radar graphs presented in Figs. 8 and 9 also show the standard deviation for each ToD over the 12-weeks. The activity, as represented by the number of temporal clusters at different times of day, in the radar graphs for P1, P2, and P3 are more similar to each other, while the radar graph for P4 shows different activity levels at different times of day for both Mondays and Thursdays. This correlates with the differences in the dispersion pattern of points in the UMAPs from Fig. 5.

When comparing Mondays to Thursdays in Figs. 8 and 9, Mondays indicate a more regular routine than Thursdays. This is also corroborated from the lower standard deviation for the majority of the ToD clusters for Mondays when compared to Thursdays. This relates to why the UMAP for Thursdays is more spread out and less distinct when compared to Mondays.

For both Mondays and Thursdays, the standard deviation for P4 is the highest (reaching a maximum of 2.21 and 2.13, respectively). The radar graphs show a more varying pattern of activity for different times of the day during P4 than during the previous periods P1, P2 and P3. The UMAPs for Thursday also indicate differences between the first three periods and P4. It should also be noted that as can be seen on the P4 radar graph for Thursdays, there are two ToD's with a standard deviation higher than 2, which could explain why the UMAP for Thursdays in P4 is much less distinct in terms of morphology and distribution.

This analysis goes some way in explaining how the UMAPs in Fig. 5 encapsulate information about the regularity of a user's routine, as when the user has a more fixed and repeatable routine, the corresponding UMAPs show a more distinct morphology and dense dispersion pattern of points. It must, however, be noted that the UMAP encodes more information from the temporal clusters than the one parameter shown in the radar graphs, as the UMAPs are generated using the full feature set as presented in Sect. 4.2. Therefore, while the radar graphs show the total number of temporal clusters for each ToD, the morphology and dispersion density of points in the UMAP plots encapsulate much more information than just the temporal clusters. Visualising the activity data through UMAP is put forward as a visualisation technique which could enable carers to identify changes over time. It is envisaged that if a visually discernible change was noted, the next step would be for the carer to examine the specific activity data in more depth and initiate informed discussions with the service user to understand what may be causing these changes and suggest appropriate interventions if the change is detrimental to their well-being. For further objective analysis of the data, pattern recognition and blob analysis to automatically detect changes in the user's routine based on the changes in the morphology and density patterns of the UMAP plots could be carried out. This would allow the system to then automatically flag changes in the user's routines as well as notify the user and their carer.

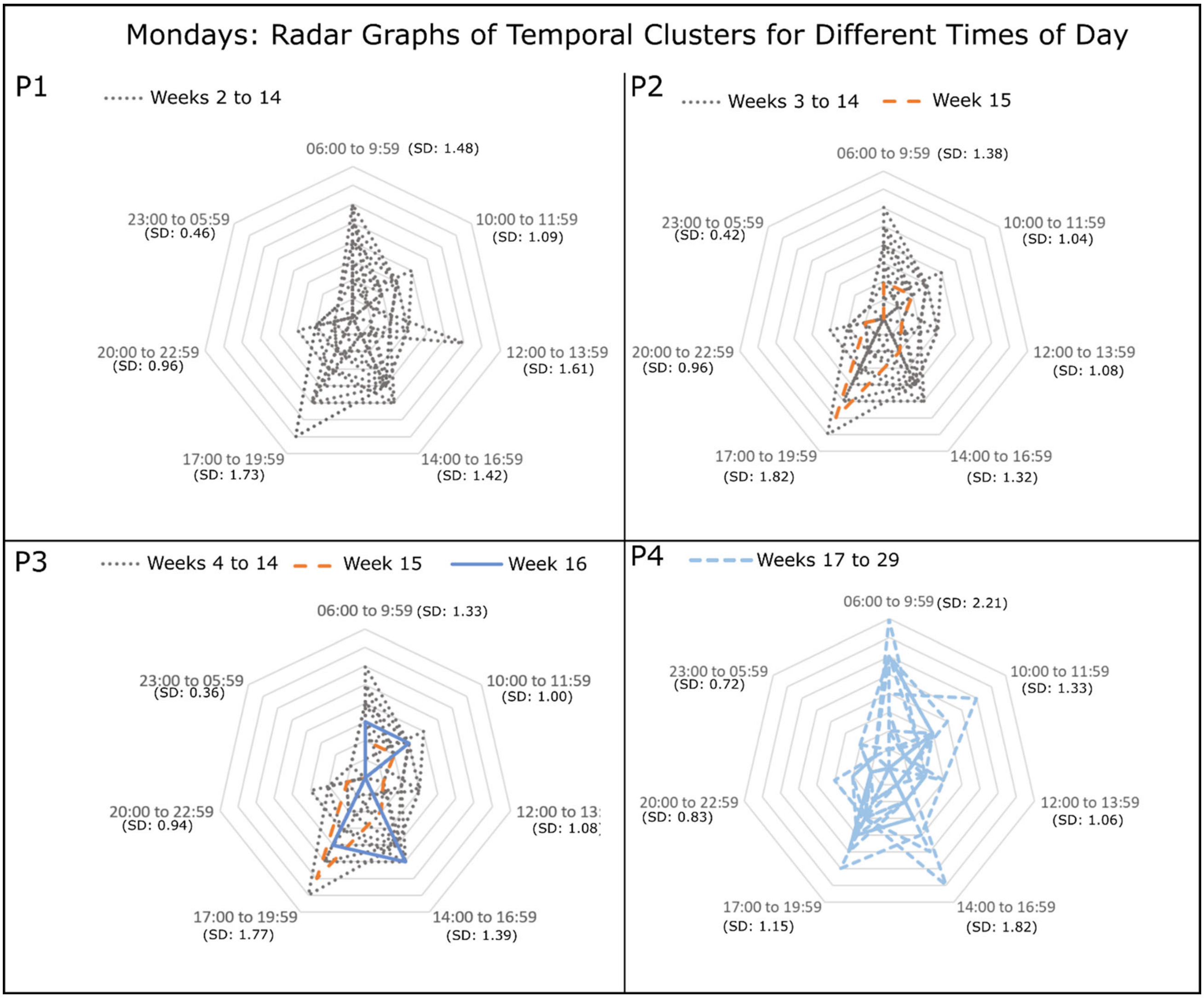


Fig. 8 Mondays: Radar graph for each 12 week period showing the total number of temporal clusters at different times of day. SD = standard deviation in total number of temporal clusters for that time of day. P1, P2, P3 and P4 refer to the four time periods included in the analysis

6 Conclusions and future work

This paper illustrates how unsupervised learning techniques can be used to discover activity patterns in unlabelled data from WSNs such as PIR sensors. A key advantage of this methodology is that it does not require hand tuning of parameters for the unsupervised learning methods. KDE is used for automatically extracting periods of dense sensor activity, as opposed to using of traditional fixed length sliding time and sensor windows. The benefit of using KDE is that the parameters can be statistically derived from the data and the method is not reliant on a fixed time window set by the user.

As carers are already overworked and have limited time for each user, it is crucial that the time they spend with the service user is utilised efficiently. The work presented in this paper revealed through UMAP and KDE, that individual week-day data, considered over long periods, could contain unique features that can be used to infer user activity levels and track any changes over the long term. The information discovered through UMAP visualisations could be further utilised as part of a structured process or assessment protocol which helps to identify anomalies or changes in user activity. This could then be used for supporting carer–patient interactions, or even tracking the effectiveness of interventions and medication on the user's health condition as indicated by their activity or changes to routines over time.

As one of the noted limitation of this study is that it is based on a single user's data, it would be important to test the methodology presented on a larger number of users, acknowledging that the method of relying on motion

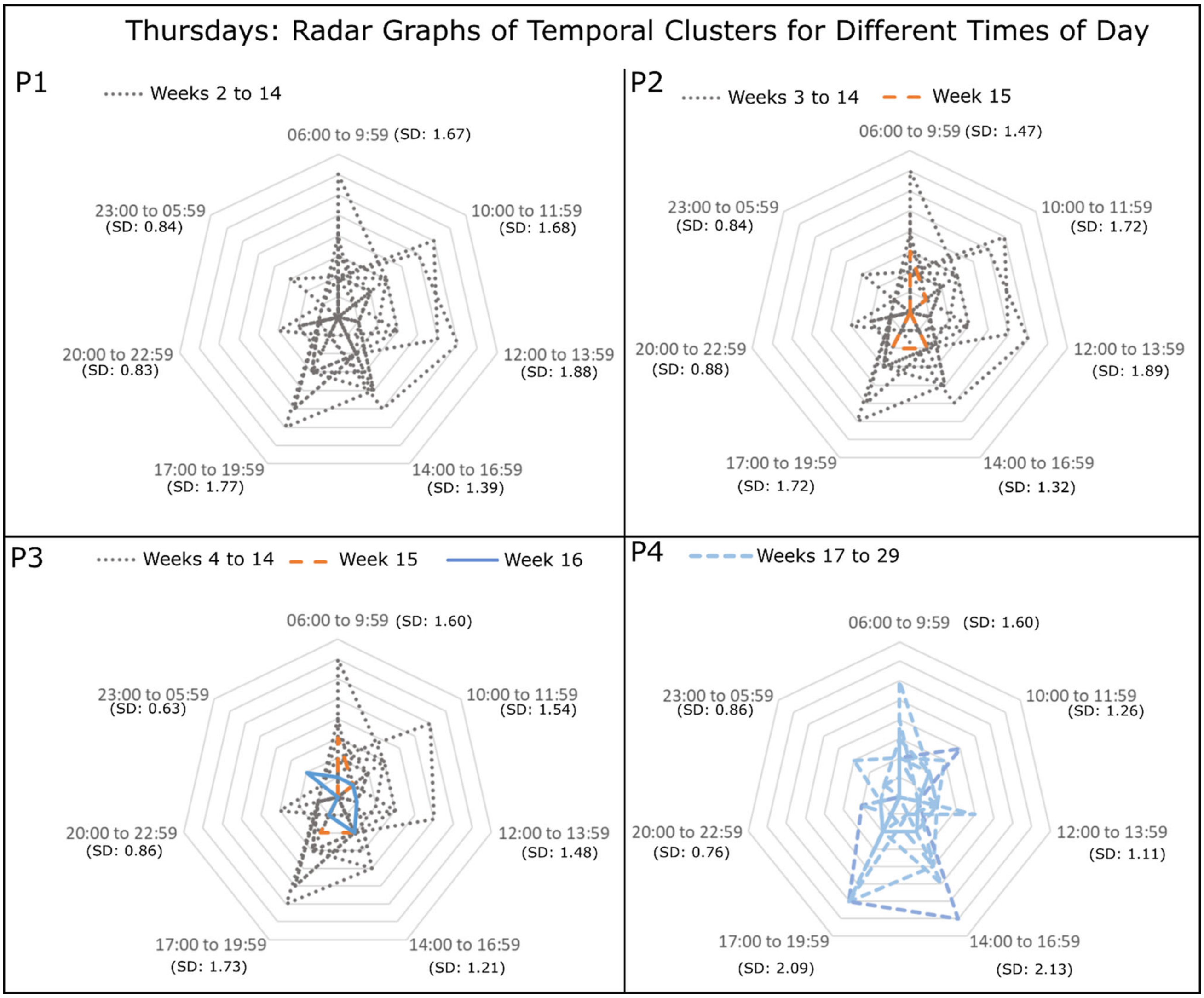


Fig. 9 Thursdays: Radar graph for each 12 week period showing the total number of temporal clusters at different times of day. SD = standard deviation in total number of temporal clusters for that time of day. P1, P2, P3 and P4 refer to the four time periods included in the analysis

sensors might not be able to track an individual's activity in a multiple occupancy scenarios, unless additional sensing is used to track an individual occupant as well.

This paper presented a novel approach to generate actionable information and insights on changing user activity over time from unlabelled data in an unsupervised manner. Future work will involve developing a real-time implementation using the KDE temporal cluster extraction technique, as well as testing on other datasets, and trialling UMAP and radar graph visualisations based in the real-world with carers and their service users. The use of pattern recognition and blob analysis will also be investigated to automatically detect and flag changes in the UMAP plots over time in order to generate notifications to the user as well as their carers. The underlying aim of this work is to develop a system that can support the user, as well as their carers, by providing actionable information based on learning their activities and routines and tracking any changes to these, without the need for labelling large amounts of data or the use of intrusive devices such as microphones and cameras.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Kvedar J, Coye MJ, Everett W (2014) Connected health: a review of technologies and strategies to improve patient care with telemedicine and telehealth. Health Aff (Millwood) 33:194–199. https://doi.org/10.1377/hlthaff.2013.0992
2. Blasco R, Marco Á, Casas R et al (2014) A smart kitchen for ambient assisted living. Sensors (Switzerland) 14:1629–1653. https://doi.org/10.3390/s140101629
3. Bauer R, Steiner M (2005) Injuries working together to make Europe a safer place in the European Union Statistics Summary 2005–2007. Report injuries working together to make Europe a safer place in the European Union Statistics Summary 2007
4. Cardinaux F, Bhowmik D, Abhayaratne C, Hawley MS (2011) Video based technology for ambient assisted living: a review of the literature. J Ambient Intell Smart Environ 3:253–269. https://doi.org/10.3233/AIS-2011-0110
5. Liu L, Stroulia E, Nikolaidis I et al (2016) Smart homes and home health monitoring technologies for older adults: a systematic review. Int J Med Inform 91:44–59
6. Boise L, Wild K, Mattek N et al (2013) Willingness of older adults to share data and privacy concerns after exposure to unobtrusive home monitoring. Gerontechnology 11:428–435. https://doi.org/10.4017/gt.2013.11.3.001.00
7. van Kasteren T, Noulas A, Englebienne G, Kröse B (2008) Accurate activity recognition in a home setting. ACM Press, New York
8. Fiorini L, Cavallo F, Dario P et al (2017) Unsupervised machine learning for developing personalised behaviour models using activity data. Sensors 17:1034. https://doi.org/10.3390/s17051034
9. Cook DJ (2012) Learning setting-generalized activity models for smart spaces. IEEE Intell Syst 27:32–38. https://doi.org/10.1109/MIS.2010.112
10. Gupta P, Caleb-Solly P (2018) A framework for semi-supervised adaptive learning for activity recognition in healthcare applications. In: Communications in computer and information science. Springer, Cham, pp 1–12
11. Rosenblatt M (1956) Remarks on some nonparametric estimates of a density function. Ann Math Statist 27(3):832–837
12. Silverman BW (1986) Density estimation for statistics and data analysis, vol 26. CRC Press, p 45
13. van der Maaten L, Hinton G (2008) Visualizing Data using t-SNE. J Mach Learn Res 9:2579–2605. https://doi.org/10.1007/s10479-011-0841-3
14. McInnes L, Healy J, Melville J (2018) UMAP: uniform manifold approximation and projection for dimension reduction
15. Becht E, Dutertre C-A, Ginhoux F, Newell E (2018) Evaluation of UMAP as an alternative to t-SNE for single-cell data. bioRxiv 298430. https://doi.org/10.1101/298430
16. Wattenberg M, Viégas F, Johnson I (2017) How to use t-SNE effectively. Distill. https://doi.org/10.23915/distill.00002
17. Yala N, Fergani B, Fleury A (2015) Feature extractionand incremental learning to improve activity recognition on streaming data. In: 2015 IEEE international conference on evolving and adaptive intelligent systems, EAIS 2015, pp 1–8
18. Epanechnikov VA (1969) Non-parametric estimation of a multivariate probability density. Theory Probab Appl 14:153–158. https://doi.org/10.1137/1114019
19. Jones E, Oliphant T, Peterson P (2001) SciPy: open source scientific tools for python—reference Guide, 0.14