2021-06-01

Monitoring Quality of Life Indicators at Home from Sparse and Low-Cost Sensor Data.

Dympna O'Sullivan
Technological University Dublin, dympna.osullivan@tudublin.ie

Rilwan Basaru
City, University of London

Simone Stumpf
City, University of London

See next page for additional authors

Authors
Dympna O'Sullivan, Rilwan Basaru, Simone Stumpf, and Neil Maiden

Monitoring Quality of Life Indicators at Home from Sparse and Low-Cost Sensor Data

Rilwan Basaru[1] Dympna O'Sullivan[2] Simone Stumpf[1] and Neil Maiden[1]

[1] City, University of London, London, UK.
[2] Applied Social Computing Research Group, School of Computer Science, TU Dublin, Dublin, Ireland.
remilekun.basaru.1@city.ac.uk, dympna.osullivan@tudublin.ie, simone.stumpf.1@city.ac.uk, neil.maiden.1@city.ac.uk

Abstract. Supporting older people, many of whom live with chronic conditions or cognitive and physical impairments, to live independently at home is of increasing importance due to ageing demographics. To aid independent living at home, much effort is being directed at reliably detecting activities from sensor data to monitor people's quality of life or to enhance self-management of their own health. Current efforts typically leverage smart homes which have large numbers of sensors installed to overcome challenges in the accurate detection of activities. In this work, we report on the results of machine learning models based on data collected with a small number of low-cost, off-the-shelf passive sensors that were retrofitted in real homes, some with more than a single occupant. Models were developed from the sensor data collected to recognize activities of daily living, such as eating and dressing as well as meaningful activities, such as reading a book and socializing. We evaluated five algorithms and found that a Recurrent Neural Network was most accurate in recognizing activities. However, many activities remain difficult to detect, in particular meaningful activities, which are characterized by high levels of individual personalization. Our work contributes to applying smart healthcare technology in real-world home settings.

Keywords: Activity recognition. Sensors, Machine Learning, Independent Living.

1 Introduction

Activity recognition is an essential component of at-home health monitoring. An understanding of a person's activities and the extent to which activities are being achieved or not can be used to improve self-monitoring and self-care at home, including their quality of life [1]. However, there are two main challenges to implementing activity recognition at home into everyday practice. First, there is the challenge of retrofitting residences with sensors. Typically, smart home solutions have hundreds of sensors with the aim of collecting data to recognize a range of different activities. The cost and complexity of such installations often prevents their take-up in real-world applications, especially if a patient is to remain in their existing residence. Second, even with large amounts of sensor data, there are challenges to developing machine learning models for activity recognition. These include noisy sensor data and large numbers of false positives, for example, a family pet will activate motion sensors. Another challenge is a lack of sensor datasets upon which to train and test activity recognition models including the fact that data collected in one home is often not useful for training algorithms designed to detect activities in another home with a different layout. A further difficulty is the multiple-occupancy problem where if more than one person is living in a home, passive sensors cannot detect which occupant is activating the sensors. In addition, research has typically focused on detecting activities of daily living (ADLs), which are tasks that people undertake routinely in their everyday lives, for example, eating, sleeping and grooming [2]. Whilst ADL recognition is well established, there is less research on monitoring meaningful activities, i.e. physical, social, and leisure activities that provide the patient with "emotional, creative, intellectual, and spiritual stimulation" [3] as an important indicator of quality of life.

To address these challenges, we investigated a toolkit composed of a small number of low-cost off-the-shelf passive sensors, typically up to 10, which were retrofitted into real, sometimes multiple-occupancy homes to detect both ADLs and meaningful activities. We collected data from five users in five different homes, each over a period of one week. We used this data to train five machine learning algorithms and evaluated their accuracy in recognizing ADLs and meaningful activities. Our work can contribute to implementing low-cost AI solution into everyday healthcare.

The rest of this paper is organized follows. First, we present an overview of current work research in activity recognition from sensor data in a home care setting. We then present the methods employed in this study, including how we collected data and ground truth labels from human participants, and how we trained and evaluated the machine learning models. We present our results, focusing on the overall accuracy of the machine learning models as well as accuracy in recognizing individual activities. We conclude by discussing the potential implications of our work, as well as directions for future research.

2 Background

Smart homes attempt to bridge the gap between monitoring and eHealth, by creating living environments which can monitor and detect behavioral patterns and disease progression of the occupants with typical approaches consisting of many hundreds of high cost sensors [4]. Approaches can be broadly categorized into passive or on-body sensing. On-body sensors are attached or carried on the user's body including smartphones, smartwatches, accelerometers and gyroscope. On body sensing data from a single or limited number of data sources often achieves high accuracy when compared to passive sensing approaches However, on-body sensors have several drawbacks including high costs, the fact that they are noticeable and invasive and have high power demands requiring frequent charging. In addition, the data these sensors collect is often proprietary (e.g. from devices such as an Apple Watch or Fitbit) and thus difficult to access for monitoring purposes. As such, others have used external sensing of the environment (so-called passive sensing) for classification of activities of daily living. Passive sensors are not worn but are placed pervasively within the residents' home environment, and are used to collect information on an occupant's daily living regimen, such as how often the resident showers, eats, and when they go to bed [5] [6] [7]. For example, in [5] Fang et al. trained a Neural Network to recognize activity based on selected featured from motion sensor events in the home. In [6], Cook et al. proposed the automated detection of frequent routines (that are then represented as an activity) that are resolved from observed patterns in signals of sensors (including motion, door and temperature) around the home. Experiments on a combination of clustering and HMM model proposed in this work show that they were able to recognize 73.8% of activities. Emi and John [7] address the multiple occupancy problem by adding microphones to a sensor toolkit in order to recognize individual occupants with a detection accuracy of over 90%.

A further form of external sensing is activity tracking based on object interaction. One commonly used technology is Radio Frequency Identification (RFID). For example, in [8], Yegang used RFID tags to detect object usage by attaching them to everyday objects such as chairs and toothbrushes to measure ADLs. They reported an accuracy of 78.3% in detecting ADLs. Beacon sensors are emerging as a low-cost alternative to RFID for activity recognition. Beacons sensor, i.e. small devices that broadcast packets of data over Bluetooth, are placed on objects in the home that residents interact with frequently. This allows capturing more minute details on a resident's activities. For example, in [9] the authors demonstrated how accelerometer data captured from beacon sensors could detect not only the presence of residents interacting with the objects, but also the way the objects were moved (e.g. placing a knife on the table vs. using the knife to cut food in its preparation) to provide finer detail about the activities performed. The results showed that simple detection classification on the manipulations of objects was able to provide 93% detection of relevant object manipulations, such as drinking from a water bottle or removing pills from their box. Niu et al. [10] propose a similar approach using BLE (Bluetooth Low Energy) beacons to measure movement and achieved an accuracy of 70% averages across seven ADLs.

There are challenges with the use of such small sensors affixed to objects. In addition to the size constraints of the sensors themselves, energy consumption can be a problem, as analyzing accelerometer data requires a high transmission rate in order to capture the movements effectively with machine learning techniques. However, accuracy of detection using beacons is relatively high and they are suited to multiple occupancy environments as they can provide specific location accuracy allowing to identify who is interacting with the device.

3 Methods

3.1 Data Collection

We recruited 5 participants (3 males, 2 females), all aged 18 and above, without any cognitive or physical impairments to take part in a pilot study. Recruitment was through convenience snowball sampling, advertised through university mailing lists and public websites. No incentives were provided. We received ethics approval prior to commencing the study and obtained informed consent from all participants. Participants were able to choose one from a set of activities (Table 1), agreed between the researcher and each participant, with a mixture of ADLs and meaningful activities. Participants carried out a set of activities over the course of one week in their own homes. They were instructed to carry out as many of these activities in their chosen set as they could over one week.

A variety of sensors were used during the data collection task. To detect interaction with objects around the home six main sensor types were used - motion, door, power, ambient (temperature and humidity), pressure and beacon sensors.

Table 1. Set of activities (meaningful activities are in italics)

Set 1	Set 2	Set 3	Set 4	Set 5
Food Preparation	Laundry	Food Preparation	Sleeping	Laundry
Meal- time	Meal- time	Bathing	*Watch TV*	Wash dishes
Bathing	Wash dishes	*Watching TV*	*Going out*	Dressing
Read Book	*Board games*	*Walk the Dog*	Shaving	*Use computer*
Board games	*Walk the Dog*	Nailcare	*Sports*	*Going out*
Housework	*Gardening*	*Gardening*	Housework	*Sports*

The motion, door, and pressure sensors are binary sensors that can detect motion in an environment, for example, opening of a door and the application of a pressure on a surface such as a bed respectively. The temperature, humidity, and power sensors are continuous sensors that detect changes in temperature, humidity and power surges. Finally, the beacon sensor is a binary sensor that detects the disturbance of any object or surface it is attached to. For example, they were attached to bookmarks and the remote control for the TV. Based on the selected set of activities, the appropriate set of sensors was provided and installed by the researcher, who noted down the location on a rough sketch of the floor plan of the participant's home. During the study, data collected from the sensors was stored on a database on a Raspberry Pi. Because of the time-dependent nature of the data being stored, we used InfluxDB [11], an open-source time series database framework, optimized for fast, storage and retrieval of time series data. The motion, door and ambient sensors used were from the same manufacturer, Xiaomi [12] and consequently had the same interfacing hub. The pressure and the power sensors were interfaced with the Raspberry Pi, using a z-wave communication protocol using a z-wave USB hardware [13]. We used the Home Assistant open-source framework [14] as a service for asynchronously listening for sensor readings and updating the InfluxDB database. The vibration sensors used in our set up broadcasts sensor reading on BLE signals. This is detected by the Bluetooth dongle that comes with the Raspberry Pi. BLE signal observed by this dongle is parsed and communicated to the Home Assistant framework using MQTT messaging protocol [15]. A typical kit was composed of 25 sensors and cost on average £412 including hub components.

Data collection took place over the course of February and March 2019. During the study, participants recorded a log of activities using a journaling app called ATracker [16] on an Android tablet to record the start and the end time of the different activities as they were completed. These logs were used as "ground truth" labels for the sensor data. After the data collection period, collected sensor data and the journaling data were jointly reviewed by the participant and a researcher, with the aim of validating the completed data set.

3.2 **Activity Dataset**

The data gathered contained a mixture of sensors readings and associated labels. There were two main challenges to overcome when using the dataset for training and evaluation. First, the output signal from different sensors was heterogeneous. For example, some of them produced binary outputs (beacons, pressure and motion sensors) whilst others (temperature, humidity, and power sensors) produce continuous outputs that vary significantly in range. We overcame this challenge by converting binary signals to a continuous format (see section 3.3 for more details). Second, the model developed for activity recognition should be invariant to the relative physical positioning of different sensors in the home, in order to allow the model to be applied in different living and sensor location configurations. To address these challenges, we propose the concept of a measurement, where a measurement is defined as a combination of a type of sensor and a type of interaction. For instance, 'beacon_sports_1' is the measurement from a beacon sensor attached to the participant's sports gear, whilst 'motion_kitchen_1' is the motion sensor in the kitchen (see Table 2 for a list of measurements). This reduces the machine learning task to establishing the mapping between measurement and activity detection. Note that this is conceptually different to established approaches where the setup and positioning of the range of sensors is fixed. The measurements and sensors they are derived from are shown in Table 2.

Table 2. Measurements, sensor types and type of interaction

Measurement Names	Sensor Type	Description
beacon_sport_1	Vibration	Vibration sensor attached to sports shoe
beacon_sport_2	Vibration	Vibration sensor attached to sports jumper
beacon_keys_1	Vibration	Beacon attached to keys
beacon_tvremote_1	Vibration	Beacon attached to tv remote control
beacon_book_1	Vibration	Beacon attached to book marker
beacon_nail_1	Vibration	Beacon attached to nail care set
pressure_computerchair_1	Pressure	Pressure sensor under computer chair
pressure_tvchair_1	Pressure	Pressure sensor under living room sofa
pressure_bed_1	Pressure	Pressure sensor under bed
pressure_mealchair_1	Pressure	Pressure sensor under reading chair
power_tv_1	Power	Power sensor attached to tv
power_washing_1	Power	Power sensor attached to washing machine
power_kettle_1	Power	Power sensor attached to kettle
door_clothe_1	Door	Door sensor attached to door of clothing cabinet
door_food_1	Door	Door sensor attached to the door of food cabinet
temp_bath_1	Temperature	Temperature sensor located in bathroom
humid_bath_1	Humidity	Humidity sensor located in bathroom
motion_bath_1	Motion	Motion sensor located in bathroom
Motion_meal_1	Motion	Motion sensor located in dining area

We collected data for 14 activities out of our initial set of 18, since none of the participants had a dog or garden or recorded using the computer as a meaningful activity (Table 1). There was high variation in the frequency and the duration of completing each task. Sleeping, for example, was recorded the most frequently (11 times) and recorded the most (95.34 hours), followed by Going Out (10 times, 30.44 hours). Food preparation was the most frequently recorded activity (24 times) but on average took much less time to do (0.29 hours). On the other end of activity frequency and duration were Vacuuming (3 times), Nail Care (2 times), Grooming (2 times), Laundry (3 times) and Playing Board Games (1 times); these activities only happened infrequently and also were recorded the least amount of time overall. To reduce bias in the subsequent prediction model (such that models developed would not be biased towards classes with higher frequency or duration), we took two actions. Firstly, we removed infrequent activities where there are not enough training and testing data (playing board games). We also applied a class weight to "boost" activities with lower frequencies. This approach is described in Section 3.3.

3.3 Model Development and Evaluation

To extract a more homogenous signal from the raw sensor readings, we applied a log function to measurements from non-binary power, temperature and humidity sensors in order to convert these to a continuous format. In addition, features must also incorporate temporal information. This was computed using the time elapsed between a point at the time of interest and the last time a sensor was triggered. Features are stored as a vector of measurements and their temporal information and each feature vector contains a label applied by the user via the ATracker app during data collection. Formally, we established a measurement feature vector, f at time t to be

$$f_t = [m_1^t, t - time(m_1^t), \dots, m_K^t, t - time(m_K^t)]\ , \qquad (1)$$

where mkt is the sensor reading, k is when it was last triggered prior to time t and time(mkt) is the time the reading, mkt was observed and K is the number of readings. Note that the time difference is observed in hours, calculated to the nearest second.

We established a target space by assuming mutual exclusivity of the different activities, i.e. only a single activity can be completed at a single time. This is consistent with the data we collected. At given time, t, a target vector a_t is established as

$$a_t = [I_t(A_1), \dots, I_t(A_C)], \qquad (2)$$

where $I_t(A_1)$ is an indicator function that yields 1 if the activity A_c a is being labelled to be done at time t and 0 otherwise. Observe that because of the mutual exclusivity assumption, $\sum_c^C \quad I_t(A_c) = 1$ at any time t. Also note that a consequence of this method of feature extraction is that sequential information, i.e. information of the about previous state of the sensors are taken into consideration by the learning model.

We evaluated five machine learning models for activity recognition. These include Naïve Bayes, a Perceptron, a Support Vector Machine (SVN), Logistic Regression trained with a Passive-Aggressive algorithm, and a Recurrent Neural Network (RNN).

Some activities were more frequently completed and take longer to complete (see Table 3). To avoid bias in the model prediction, we applied a weighting on the contribution to the summative loss of all data points in a training batch. Specifically, during training of the models, the computed loss of each data point is weighted with the inverse its class total duration (see Table 3). The effect of this is that the loss computed for data point of a class of activity with higher total duration (like 'Sleeping') will be allocated a lower weighting.

Table 3. Duration and frequency of activities (meaningful activities are in bold)

Activity	Total Duration (Hours)	Freque ncy	Mean Duration (Hours)
Sleeping	95.3496	11	8.6681
Going out	30.4447	10	3.0445
Watching TV	10.4795	14	0.7485
Food prep	7.0502	24	0.2938

Mealtime	5.3089	14	0.3792
Reading	3.0964	11	0.2815
Washing dishes	2.6332	15	0.1755
Bathing	1.0304	9	0.1145
Dressing	0.838	6	0.1397
Housekeeping (vacuum)	0.6936	3	0.2312
Nail care	0.4192	2	0.2096
Grooming	0.25044	2	0.1252
Laundry	0.2299	3	0.6897

We used the ScikitLearn [62] Python machine learning library to implement the SVM, Naïve Bayes, Logistic Regression, and Perceptron models. The Naïve Bayes was multinomial and was trained with an adaptive smoothing parameter (alpha) of 0.01. The SVM model was trained with 5 maximum epochs. The Perceptron model was trained with a stopping criterion of 1e-3. Unlike the first 3 models the RNN was implemented with the TensorFlow framework [63] and trained with a learning rate of 0.001, weight decay of 0.005 and under 2 epochs. The above training parameters were chosen after experimenting with several parameters. Data was split into training and validation sets by a 75:25 ratio, respectively.

The performance of the machine learning models was measured by comparing predicted activities with ground truth activities gathered via the Atracker app using several metrics. We calculated accuracy for each algorithm as a ratio of all correctly labelled data point to all test data points. Further, we computed precision (the fraction of all detected activities that are actual activities), recall (the fraction of all activities that are successfully detected), and the F1-score (the harmonic average of precision and recall and accuracy which is the percentage of correctly classified activities). To take into consideration the imbalanced nature of the data, we also computed micro and macro averages for precision, recall, and F1-score. Micro-average averages the metrics across all data points. Macro-average entails first computing the metrics for each class independently before then taking an average; hence it addresses any potential class imbalance in the data.

4 Results

4.1 Model Accuracy

We first investigated the overall performance of each algorithm. RNN achieved the highest average accuracy across all evaluated activities, correctly recognizing 65.59% of the activities from the dataset followed closely by Perceptron on 65.09%. The other models performed as follows - SVM (59.3), Logistic Regression (58%) and Naïve Bayes (53.95%).

The micro-average, macro-average and F1 scores for each classifier and shown in Table 4 give further insight into the results. Here, the RNN yields the highest scores, with macro-average precision and recall scores of 0.88 and 0.41 respectively, and a F1-score of 0.46. In fact, it significantly outperforms the other classifiers at correctly recognizing a range of activities, achieving a macro average F1-score that is 228.5% higher than the Perceptron. We further explored the superiority of the RNN model to the other models by computing the McNemar test between the correctly predicted instances by the RNN model and by the other four model with alpha = 0.05. The test indicated that the prediction performance of the RNN was statistically significant with all four-comparison yielding a p-value of 5.2×10-12, 8.2×10-14, 6.2×10-16 and 3.2×10-17 against Perceptron, Logistic Regression, Naïve Bayes and SVM respectively. We hypothesize that the superiority of RNN is owed to its inherent feedback architecture, which allows it to hold latent information about the previous state of the model in memory. Although we extracted features using time windows (as discussed in Section 3.3) and as such the other models are also exposed to information about the previous state of the system, this is limited. One reason for this could be the fixed time window might not be large enough, or it might be too long to encapsulate the essence of the activity. For example, looking at Table 3 the Sleeping activity is typically completed in 8 hours, hence a suitable time window for tracking the Sleeping activity will be too large for tracking Laundry (which typically takes 40 minutes). The RNN model is better able to adjust its weight (during the training step) to adaptively retain information over time.

Table 4. Micro-averaged and macro-averaged precision, recall and F1-scores

Algorithms		Precision	Recall	F1-Score
SVM	Micro Average	0.59	0.59	0.59
	Macro Average	0.18	0.10	0.09
Naïve Bayes	Micro Average	0.54	0.54	0.54
	Macro Average	0.04	0.07	0.05
Logistic Regression	Micro Average	0.58	0.58	0.58
	Macro Average	0.11	0.10	0.09
Perceptron	Micro Average	0.65	0.65	0.65
	Macro Average	0.16	0.14	0.14
RNN	Micro Average	0.56	0.56	0.56
	Macro Average	**0.88**	**0.41**	**0.46**

4.2 Activity Accuracy

We explored the performance of the models across the different activities (Table 5). SVM only recognizes three activities (No Activity, Sleeping, and Going Out), while Logistic Regression can distinguish between two activities (No Activity and Going Out). Naïve Bayes only recognizes when there is No Activity i.e. it can detect when nothing is done but it cannot accurately predict what is done. Perceptron had high overall accuracy and also a high micro-average accuracy, however, the macro-average accuracy showed that it is not very good at recognizing a variety of activities. As can be seen in Table 5, it can recognize only three activities: No Activity, Watching TV, and Going Out. In comparison, the RNN can recognize a much wider range of activities reliably than the Perceptron.

Our results also highlight activities that are problematic to recognize reliably. We found that seven activities had low F1-scores across all algorithms, i.e. none of the approaches we tried worked very well for recognizing Washing Dishes, Mealtime, Food Prep, Watching TV, Sleeping, Reading and Grooming. Note that all meaningful activities are in the bottom half of Table 4, meaning that these kinds of activities seem to be most troublesome to recognize.

Table 5. F1-scores per activity, decreasing order of RNN's F1 score (Meaningful activities in bold).

	SVM	Naïve Bayes	Logistic Regression	Perceptron	RNN
Nailcare	0.00	0.00	0.00	0.00	1.00
Laundry	0.00	0.00	0.00	0.00	0.98
Housekeeping	0.00	0.00	0.00	0.00	0.85
Bathing	0.00	0.00	0.00	0.00	0.82
Mealtime	0.00	0.00	0.00	0.00	0.22
Dressing	0.00	0.00	0.00	0.00	0.75
No Activity	0.72	0.70	0.72	0.76	0.71
Wash Dishes	0.00	0.00	0.00	0.00	0.46
Food prep	0.00	0.00	0.00	0.00	0.14
Watching TV	0.00	0.00	0.00	0.31	0.11
Sleeping	0.16	0.00	0.00	0.00	0.04
Going Out	0.40	0.00	0.51	0.88	0.04
Reading	0.00	0.00	0.00	0.00	0.00
Grooming	0.00	0.00	0.00	0.00	0.00

5 Discussion and Conclusions

Our results are comparable to others using external sensing approaches that use a larger amount of sensors, e.g. [5]. Of particular interest is that an RNN model shows promise given that we have used a limited number of cheap off-the-shelf sensors and a very low number of training examples when compared to previous work. Furthermore, we have focused on more difficult to detect meaningful activities in addition to ADLs and used data collected from various setups and locations corresponding to real homes.

Our results highlight that the models suffered from class imbalance and that many activities were difficult to recognize. We believe this is because many of the activities are not singular, rather they involve a number of distinct subtasks, many of which are not crisply defined, e.g. meal times may involve laying a table with cutlery or plates and sitting at a table or alternatively it can involve eating food in front of the TV. Furthermore, real users may have different routines for different mealtimes, for example, breakfast may be a faster event and involve fewer tasks that eating dinner. This suggests careful consideration needs to be given to the set and combination of sensors to capture these activities. Furthermore, high levels of personalization are likely to be necessary for detecting meaningful activities, which can be learned from collecting and studying datasets collected over longer periods to analyze user habits.

In future work we are interested in addressing our limitations in using BLE sensors. We propose the use of conventional Bluetooth, as opposed to BLE, although this will consume more energy, they can be detected by sensors on mobile devices more consistently. This can help detect location and hence solve for activity recognition in multi-occupancy scenarios. This may also help to with recognizing meaningful activities and their more personalized nature.

Acknowledgments

This material is based upon works supported by the Science Foundation Ireland under Grant No. 19/FFP/6917.

References

1. S. Majumder, E. Aghayi, M. Noferesti, H. Memarzadeh-Tehran, T. Mondal, Z. Pang, M.J. Deen. Smart Homes for Elderly Healthcare—Recent Advances and Research Challenges. Sensors 2017, 17, 2496.
2. M. Wallace and S. Mary, "Katz Index of Independence in Activities of Daily Living (ADL)," Urologic Nursing Journal, vol. 27, no. 1, pp. 93-94, 2007.
3. NICE, "Mental wellbeing of older people in care homes," 2013. [Online].
4. J. Ye, G. Stevenson, S. Dobson, Detecting abnormal events on binary sensors in smart home environments, Pervasive and Mobile Computing, vol no 33, pp 32-49, 2016
5. H. Fang, L. He, H. Si, P. Liu and X. Xie, "Human activity recognition based on feature selection in smart home using back-propagation algorithm," ISA Transactions, vol. 53, no. 5, pp. 1629-1638, 2014.
6. D. Cook, N. Krishnan and P. Rashidi," Activity Discovery and Activity Recognition: A New Partnership" IEEE Trans Cybern. 2013 Jun; 43(3): 820–828.
7. I. A. Emi and A. S. John, "SARRIMA: Smart ADL recognizer and resident identifier in multi-resident accommodations," in The Wireless Health Conference, Bethesda, USA, 2015.
8. D. Yegang, L. Yuto and T. Yasuo, "A Novel Human Activity Recognition and Prediction in Smart Home Based on Interaction," Sensors, vol. 19, no. 20, pp. 1-16, 2019.
9. G. Civitarese, S. Belfiore and C. Bettini, "Let the objects tell what you are doing," in The ACM International Joint Conference on Pervasive and Ubiquitous Computing, Heidelberg,Germany, 2016.
10. L. Niu, S. Saiki, L. D. Bousquet and M. Nakamura, "Recognizing ADLs Based on Non-intrusive Environmental Sensing and BLE Beacons," in The 8th International Conference on Indoor Positioning and Indoor Navigation, Sapporo, Japan, 2017.
11. InfluxDB, "InfluxDB: Purpose-Built Open Source Time Series Database," [Online]. Available: https://www.influxdata.com/. [Accessed 12 April 2020].
12. Xiaomi, "Mi Global Home," [Online]. Available: https://www.mi.com/global/mi-smart-sensor-set/. [Accessed 23 April 2020].
13. Z-Wave, "Z-Wave | Safer, Smarter Homes Start with Z-Wave," [Online]. Available: https://www.z-wave.com/. [Accessed 20 January 2020].
14. Home-Assistant, "Home Assistant," [Online]. Available: https://home-assistant.io. [Accessed 21 September 2019].
15. MQTT, "MQTT," [Online]. Available: http://mqtt.org/. [Accessed 20 September 2019].
16. Wonderapps, "ATracker - Daily Task and Time Tracking," [Online]. Available: http://www.wonderapps.se/atracker/home.html. [Accessed 19 March 2020].