



# Efficacy of Imbalanced Data Handling Methods on Deep Learning for Smart Homes Environments

Rebeen Ali Hamad<sup>1</sup> · Masashi Kimura<sup>2</sup> · Jens Lundström<sup>3</sup>

Received: 28 February 2020 / Accepted: 29 May 2020  
© The Author(s) 2020

## Abstract

Human activity recognition as an engineering tool as well as an active research field has become fundamental to many applications in various fields such as health care, smart home monitoring and surveillance. However, delivering sufficiently robust activity recognition systems from sensor data recorded in a smart home setting is a challenging task. Moreover, human activity datasets are typically highly imbalanced because generally certain activities occur more frequently than others. Consequently, it is challenging to train classifiers from imbalanced human activity datasets. Deep learning algorithms perform well on balanced datasets, yet their performance cannot be promised on imbalanced datasets. Therefore, we aim to address the problem of class imbalance in deep learning for smart home data. We assess it with Activities of Daily Living recognition using binary sensors dataset. This paper proposes a data level perspective combined with a temporal window technique to handle imbalanced human activities from smart homes in order to make the learning algorithms more sensitive to the minority class. The experimental results indicate that handling imbalanced human activities from the data-level outperforms algorithms level and improved the classification performance.

**Keywords** Activity recognition · Smart home · Imbalanced class

## Introduction

By equipping environments such as ordinary homes with binary sensors for monitoring resident activities, a vast area of different applications is made possible, including smart monitoring of energy utilization and assessing resident situation and behavior pattern for proactive home care. In the case of monitoring for home care, independent living solutions have been provided for older adults in their own homes by smart home technology to improve and maintain the quality of life and care [2, 27, 33]. Smart homes that

are used for transparently represent how, when and where humans perform activities opens up diverse health technology applications such as anomaly detection (e.g., falls) or tracking progression of diseases or recovery. Activity recognition (AR) has progressed by the recent advancement of machine learning to enhance elderly care alert systems and improve assistance in emergency situations from smart home data [12]. Another example of an application requiring AR includes smart medication reminders [40] which utilize the contexts in which to send a reminder. Similar to medication reminders is the application of assisting people with cognitive impairments to complete tasks [9]. These applications relying on AR would potentially benefit from a more accurate recognition. Moreover, by tracking the characteristics of activities related to basic needs and their change over time renders a possibility to assess parts of the progression of a persons *functional ability*, which is a focus concept for how WHO defines healthy aging. Activities of in-home mobility as showering, watching TV, cooking, eating, sleeping and grooming are therefore of importance to monitor and track in order to assess the functional health status of older adults. Moreover, the framework of AR using machine learning methods provides enough mechanisms to detect both

---

✉ Rebeen Ali Hamad  
rebeen.ali\_hamad@hh.se

Masashi Kimura  
kimura@convergence-lab.com

Jens Lundström  
jens@convergia-consulting.io

<sup>1</sup> Intelligent Systems and Digital Design, Halmstad University, Halmstad, Sweden

<sup>2</sup> Convergence Lab, Tokyo, Japan

<sup>3</sup> Convergia Consulting, Halmstad, Sweden

ambulatory and postural activities, actions of residents and body movements using different multimodal data generated by heterogeneous sensors [5, 19, 31].

Not only are human activities highly diverse in the form of different sensor activations but the frequency of activities themselves is inherently imbalanced and hence accurate AR is challenging from a machine learning perspective. Large differences in the number of examples for the classes to learn can make the machine learning algorithm to put emphasis on learning majority classes and thereby partially or completely neglect minority classes. As an example, cooking may occur with a higher frequency than grooming. Another more prominent example is the vast difference in the number of examples between eating and sleeping where the latter occurs with a much higher frequency in datasets collected over a long duration. This paper focuses on investigating the particularly problematic aspect of learning activities over days or even months which are imbalanced.

Despite many past efforts of research on the class imbalance problem and approaches to cope with this general problem, there is a lack of empirical work on targeting machine learning beyond shallow methods [20]. Traditional machine learning algorithms such as decision tree, support vector machine, naive Bayes and hidden Markov models have been used to minimize the recognition error [6, 23]. Satisfying recognition results have been achieved by adopting these approaches. However, such algorithms may heavily depend on classical heuristic and hand-crafted feature extraction which might be limited by human domain knowledge [39]. A natural variation within each activity is often present in collected smart home datasets and is not unlikely to fluctuate even more between different residents. These variations are also influenced by contextual factors such as time of the day and location of where the activity is performed. Given these conditions as well as considering the multitude of choices at sensor installation (e.g., sensor types and sensor locations), AR based on shallow learning where features are hand-crafted can be challenging. Therefore, discovering more systematic methods to obtain features has drawn increasing research interests [24]. The influence of deep learning has been demonstrated in many areas not only in image classification such as speech recognition and natural language processing as surveyed in [39]. Consequently, studies of activity recognition using deep learning have multiplied because the number of elderly smart-home healthcare services has steadily increased for the last few years and all reporting state-of-the-art performances achieved on diverse activity recognition benchmark datasets [16, 43]. Particularly, two methods have brought promising results of AR, long short-term memory (LSTM) and convolutional neural networks (CNNs) when using data prepared with a fuzzy-based approach to represent temporal components of the data [15, 26, 28]. However, to the best of our knowledge, these two

machine learning algorithms for AR have not been studied from the context of different temporal preprocessing methods along with traditional methods for handling class imbalance in order to improve recognition accuracy. The study described in this paper is therefore designed to fill parts of such a knowledge gap and also put a particular focus on the classes representing activities with a relatively low number of observations (i.e., minority classes). Thus, the main contribution of this paper is the study of well-known class imbalance approaches (synthetic minority over-sampling technique, cost-sensitive learning and ensemble learning) applied to activity recognition data with various temporal data preprocessing for the deep learning models LSTM and 1D CNN.

The rest of the paper is organized as follows. In Sect. 2, related work is described, and in Sect. 3 Methodology, the outline and details of the study are described, whereas in Sect. 4, experiment results are presented and discussed. Finally, the findings and opportunities of further research are summarized in Sect. 5, Conclusion and future work.

## Related Work

Elements of the class imbalance problem are widely studied, especially from a shallow learning perspective. Extensive work by [18] outlined three important factors of the problem: the complexity of concept (or underlying distributions), training set size and degree of imbalance. It was shown that problems with low concept complexity were insensitive to class imbalances but with an increased concept complexity the models (C5.0 & MLP) performed poorly, even when a low-class imbalance was present. Moreover, Japkowicz and Stephen concluded that a severe complex problem could be handled with a good performance given a sufficiently large amount of training data [18]. Finally, their conclusion that over-sampling and cost-modifying methods for improving model performance are preferred over an undersampling strategy, is a direction explored in this paper for deep learning models.

The intrinsic property of classes representing human activities to be imbalanced makes the topic of AR learning algorithms for imbalance handling crucial to study, especially since the arrival of deep learning which typically requires a larger dataset. Different strategies for dealing with class imbalance for deep learning were recently surveyed by [20]. The survey revealed that the number of research studies containing empirical work on targeting the class imbalance problem for deep learning is limited. However, the same survey showed that classical methods for handling imbalance (e.g., random over-sampling of minority classes and cost-sensitive target function to avoid

skewed learning toward majority classes) applied in deep learning situations show promising results.

Most past works on handling class imbalance for deep neural networks focus on computer vision tasks where image classification dominates the reviewed papers and hence not directly translatable to an AR setting. A modified cost-sensitive learning scheme was proposed by [22] with good results compared to standard cost-sensitive (when the target function is weighted toward the size or importance of classes) approaches and sampling methods (where the majority classes are undersampled or minority classes are over-sampled). However, the evaluation was based on data for image classification tasks. Another novel approach (focusing on a vision classification problem) combined sampling and a modified hinge loss to render tighter constraints between classes for a better discriminative deep representation [17]. The focus of this paper is class imbalance handling for activity recognition in a deep learning context which has earlier been approached by Nguyen et al. who proposed an extension to the random over-sampling method SMOTE called BLL-SMOTE which improved the classification results drastically [30]. However, the study was limited to mobile phone sensors which is only a subset of the type of sensors available as smart home technology.

Besides handling imbalanced activity classes, the domain of activity recognition often needs alignment to the use of a carefully selected temporal window size. In the case of mobile sensing devices, the use of a temporal window size needs a thorough analysis to properly and correctly segment the data [4]. Shallow learning schemes such as support vector machines (SVMs), decision tree or hidden Markov model based on the dynamic or sliding windows have previously been evaluated [11, 36, 38, 42]. These studies have aimed to adjust dynamic or fixed window size to enhance the performance of the classifiers. Binary stream sequence data are mostly split into subsequences called windows, where every window is related to a broader activity by a sliding window technique. Binary sensor data segmentation using only one window for deploying HAR cannot provide accurate results since the duration of human activities differ and the exact boundaries of activities are difficult to specify. Intuitively, decreasing the window size has led to increasing the performance of activity recognition in addition to minimizing resources and energy needs [4]. It has been found that the window size of 60 s extracts satisfactory features for activity recognition from smart home [26, 32].

Consequently, thorough comparisons of the use of fixed window size and fuzzy temporal windows (of particularly one hour) are important to study. The contribution of this paper is therefore significant to alleviate the complexity of defining the window size and to correctly, easily and rapidly recognize real-time imbalanced activities.

## Methodology

In this study, aspects of how to approach the class imbalance problem are considered. This section describes the relevant key components: window methods for pre-processing, machine learning algorithms used and class imbalance strategies.

### Methods to Handle Imbalanced Class Problem

The following two methods are used to handle the imbalanced class problem in activity recognition from algorithm level and data level.

#### Cost-Sensitive

Cost-sensitive is one of the commonly used algorithm level methods to handle classification problems with imbalanced data in machine learning and data mining setting [44]. Cost-sensitive evaluates the cost associated with misclassifying samples. Cost-sensitive is not creating balanced data distribution; rather, this method assigns the training samples of different classes with different weights, where the weights will be in proportion to the misclassification costs. Then, the weighted samples will be fed to learning algorithms [45].

#### SMOTE

Synthetic minority over-sampling technique (SMOTE) is a commonly used data-level method to handle imbalanced data and is based on sampling. This method over-samples the minority classes by creating synthetic samples rather than by over-sampling with replacement [7]. The minority classes will be over-sampled by selecting each minority class sample and generating synthetic observations along the line segments joining any/all of the  $k$  minority class nearest neighbor. Neighbors will be randomly chosen from the  $k$  nearest neighbors depending on the amount of required over-sampling. Commonly five nearest neighbors are used in practice. For example, if 200% is the amount needed to be over-sampled, only two neighbors are selected from the five nearest neighbors and one sample will be created in the direction of each. Synthetic samples are created by taking the difference between the sample and its nearest neighbor. The difference will be multiplied by a random number between 0 and 1 and added to the feature vector. This procedure will effectively force the decision region of the minority class to become more general. The synthetic samples will be generated in a less application-specific manner by operating in feature space instead of data space to alleviate the issues with class imbalanced distribution. Despite the common use

of SMOTE at data level, the method is less studied in deep learning contexts nor is it, to the best of our knowledge, studied together with the effect of windowing pre-processing techniques (described in section 3.3). Thereby, this paper aims to explore the potential enhancements of class imbalance approaches (where SMOTE is one of the tested methods) together with two deep learning models (1D CNN and LSTM) and several pre-processing methods described in later sections.

### Ensemble Techniques

Ensemble techniques combine several based models into one single model to enhance prediction and decrease bias and variance. The decision of several estimators on a different randomly selected subset of data will be combined to improve overall performance [14, 41]. However, commonly the subsets of data are not balanced as input to the classifiers in the ensemble. Therefore, the classifiers may favor the majority classes and generate a biased model during the training phase on the input imbalanced datasets. To overcome this problem and to reasonably compare the results of the ensemble model with the cost-sensitive and SMOTE, balanced ensemble learning is used in this study which is introduced in [13]. Balanced ensemble learning will first balance the data and then will combine the decision of multiple classifiers to avoid bias and to render better performance. Decision trees as the base models with bootstrap aggregation (Bagging) are used to build the ensemble learning.

### Smart Home Data for Evaluation

We used the activities of daily living (ADLs) for recognition using binary sensors dataset, which were acquired in two real intelligent homes A and B in which residents perform their daily routine [32]. These two homes are equipped with sensors that are able to capture the movements and interactions of the inhabitants. The binary sensors are passive infrared (PIR) motion detectors to identify movement in a specific area, pressure sensors on beds and couches to detect the user's presence, reed switches on cupboards and doors to measure open or close status and float sensors in the bathroom to measure toilet being flushed or not. The use of PIR sensors as well as pressure sensors is limited in their ability to capture details compared to other sensors such as cameras or accelerometers. However, low-resolution sensors such as PIR and pressure sensors may preserve the privacy and integrity of residents to a greater extent than for example cameras. Table 1 shows details of the two homes with information of the resident, number of activities and sensors. In home A, 9 human daily activities that were performed in 14 days over a period of 19,932 min were described by an incoming stream of binary events from 12 sensors in

**Table 1** Details of recorded datasets

|            | Home A  | Home B  |
|------------|---------|---------|
| Setting    | Home    | Home    |
| Rooms      | 4       | 5       |
| Duration   | 14 days | 21 days |
| Sensors    | 12      | 12      |
| Activities | 10      | 11      |

**Table 2** Number of observations for each activity in the datasets

| Activity       | Home A | Home B |
|----------------|--------|--------|
| Spare Time/ TV | 8555   | 8984   |
| Sleeping       | 7866   | 10763  |
| Leaving        | 1664   | 5268   |
| Idle           | 1598   | 3553   |
| Lunch          | 315    | 395    |
| Toileting      | 138    | 167    |
| Breakfast      | 120    | 309    |
| Grooming       | 98     | 427    |
| Showering      | 96     | 75     |
| Snack          | 6      | 408    |
| Dinner         | –      | 120    |

the home. In home B, ten human daily activities that were performed in 22 days over a period of 30,495 min were described by 12 binary sensors. The timeline of the activities is segmented in time slots using the window size  $\Delta t = 1$  min. The activities of homes A and B that were manually labeled are *Breakfast*, *Grooming*, *Idle*, *Leaving*, *Lunch*, *Showering*, *Sleeping*, *Snack*, *Spare Time/TV*, *Toileting*; in addition to these, home B has the activity *Dinner*.

Leave-one-out cross-validation is used and repeated this for every day and for both homes. Deep learning models (described in the next section) are trained for each home since the number of sensors varies and a different user resides in each home. Sensors are recorded at one-minute interval for 24 h, which totals in 1440 length input in minutes for each day. The average F-score is computed from the results of the cross-validation. Since the classes of the datasets are imbalanced, we propose synthetic minority over-sampling technique (SMOTE) as input data for the deep learning model. This allows us to handle the imbalanced activities and avoid having models biased toward one class or the other (Table 2).

### Data Pre-Processing

Multiple and incremental fuzzy temporal windows (FTWs) are used to extract features. Each FTW  $T_k$  is defined by a fuzzy set characterized with a membership function, and its shape corresponds to a trapezoidal function  $T_k[l_1, l_2, l_3, l_4]$ . The

well-known trapezoidal membership functions are defined by a lower limit  $l_1$ , an upper limit  $l_4$ , a lower support limit  $l_2$  and an upper support limit  $l_3$ . The values of  $l_1, l_2, l_3, l_4$  are defined by the Fibonacci sequence which was previously shown as a successful sequence for defining FTWs without requiring expert knowledge definition [15, 25]. Figure 2 shows nine windows of FTWs created based on Fibonacci sequence. To extract features, the FTWs are slid over sensors activations  $x$  in every minute according to Eq. (1): Features are computed by applying 15 FTWs on the raw data from all 12 binary sensors in each minute for both datasets. The datasets A and B have 19,932 and 30,495 samples, respectively, where each sample represents one minute of data with  $12 \times 15 = 180$  features. The resulting datasets are used for real-time activity recognition. Algorithm 1 shows the procedure of computing FTWs. Feature extraction based on FTWs is evaluated and compared with equally sized (1 min) temporal windows (ESTWs) [34] as shown in Fig. 1.

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

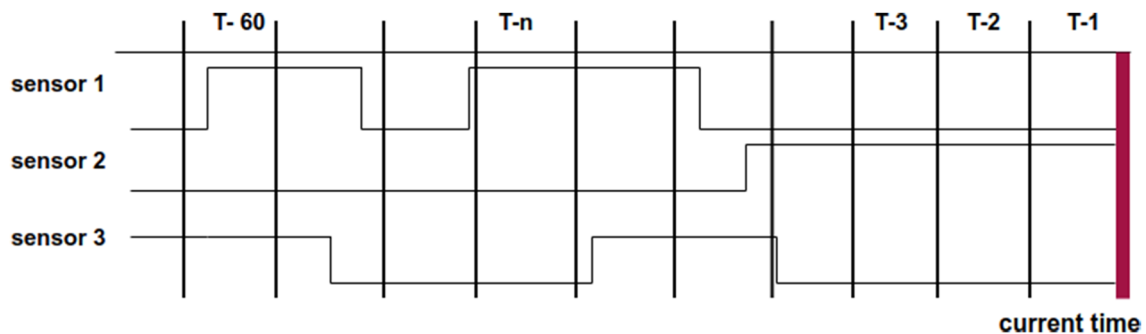


Fig. 1 Example of temporal segmentation on time series of three sensors by the equally sized temporal window method

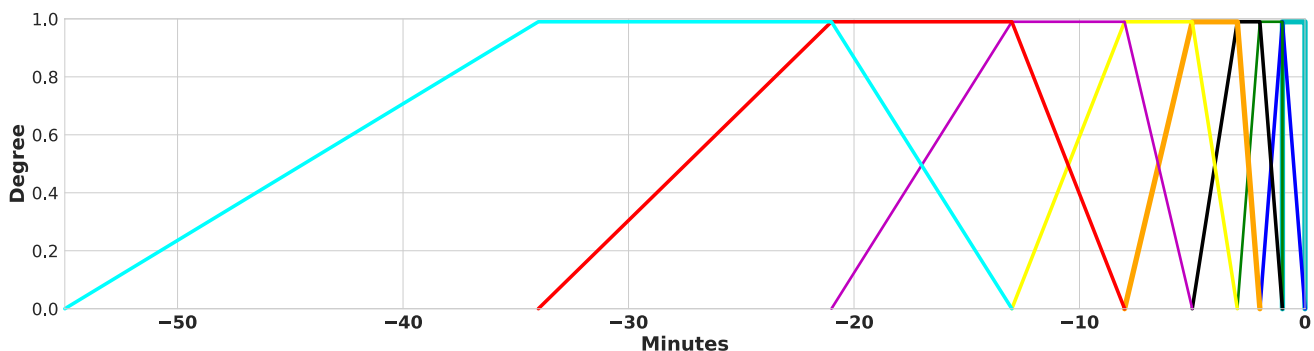


Fig. 2 Example of temporal segmentation on sensors time series by the fuzzy incremental temporal windows method

#### Algorithm 1 Extracting Features using FTWs

```

1: Input: Raw_data      Home A and B are the input Raw
                        data
2: FTWs  $\leftarrow$  Fibonacci    FTWs get values from Fibo-
                        nanci
3: Sensor_intervals  $\leftarrow$  Raw_data    sensor intervals data
4: for ftw  $\leftarrow$  FTWs do
5:   for sen_intv  $\leftarrow$  Sensor_intervals do
6:     apply ftw on sen_intv
7:   end for
8:   features  $\leftarrow$  max(ftw)
9: end for
10: dataset  $\leftarrow$  features
11: Output: dataset

```

Algorithm 2 shows the process of handling imbalanced class problem where firstly data preprocessed by FTWs or ESTWs and then infrequent classes are over-sampled by SMOTE to be used as the input data of the models (Fig. 2).



**Algorithm 2** Process of Handling imbalanced Data

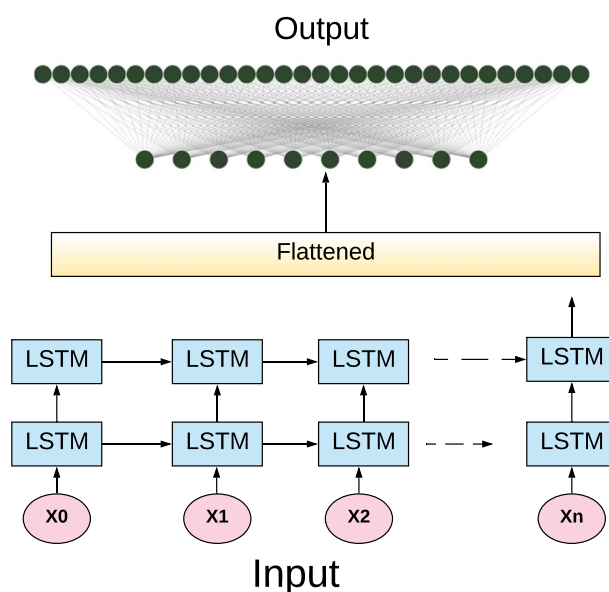
- 1: **Input:** *Raw\_data*      input Raw data
- 2: *FTW*, *ESTW*      *FTW*, *ESTW* to extract features and build datasets
- 3: *SMOTE*  $\leftarrow$  *datasets*      oversamples infrequent classes
- 4: *LSTM*, *1D CNN*  $\leftarrow$  *datasets*      Apply temporal models

**Model Selection and Architecture**

In this study, we investigate two types of neural networks: One is based on LSTM (long short-term memory) and another is based on CNN (convolutional neural network). The architecture and parameters of the temporal models are described in the following.

**LSTM**

LSTM is the extended form of the recurrent neural network (RNN) that is designated to learn from temporal sequential pattern data. We expect an LSTM architecture to handle the activity timeline of a smart home. LSTM solves the vanishing gradient problem of a simple RNN which cannot learn long-term sequences and lose the effect of initial dependencies in the sequence. LSTM is most widely used in natural language processing, stock market prediction and speech recognition that can model temporal dependence between observations [8]. LSTM has obtained satisfying results in activity recognition [16, 29]. Hence, in this study LSTM is used in the experiments by stacking two LSTM layers with 40% dropout rate and 0.001 learning rate followed by a fully connected, i.e., dense layer and softmax layer. For all the models in this study, the batch size and training epochs are equal to 10, which is a total of 100 batches during the entire training process. While large batch size commonly results in faster training, it is unable to converge as fast. On the other hand, smaller batch sizes train slower but could converge faster; therefore, it is mostly an independent problem [10]. Regarding the 40% dropout, which is a regularization technique for preventing deep learning models from overfitting [35], the dropout ignores randomly selected neurons during the training phase. Those ignored neurons are temporally removed on the forward pass and their weights are not updated on the backward pass (Fig. 3).



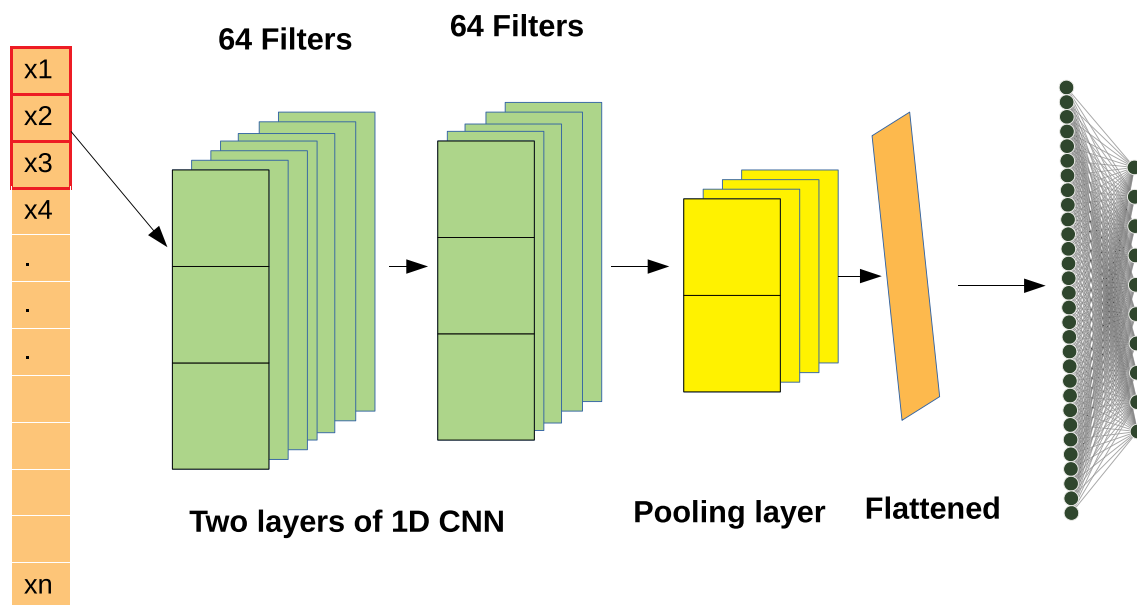
**Fig. 3** Architecture of LSTM

**1D CNN**

Convolutional neural network (CNN) is used in the experiments because it is competent in extracting features from signals. CNN has obtained promising results in image classification, text analysis and speech recognition [16]. CNN has two advantages for human activity recognition which are local dependency and scale invariance. Local dependency refers to the nearby observations in human activity recognition that are likely to be correlated, while scale invariance means the scale is invariant for different paces or frequencies. CNN can learn hierarchical data representations which lead to rendering promising results in human activity recognition [16]. In this study, a one-dimensional (1D) CNN architecture is used and can extract local 1D subsequences from the sequence data. The 1D CNN could be competitive with RNN on some sequence-processing applications such as audio generation and machine translation with a cheaper computation cost compared to RNN [3, 15]. The model is designed by stacking two convolutional layers each with 64 filters, kernel size 3 and stride 1 with 40% dropout rate and 0.001 learning rate followed by a max-pooling layer and followed by a fully connected, i.e., dense layer and softmax layer (Fig. 4).

**Measure Evaluation**

How the classification performance is evaluated plays an important role in this study. Without proper measures, no



**Fig. 4** Architecture of 1D CNN

deeper insight could be achieved. Traditionally, accuracy was commonly used to measure the performance of classifiers. However, for classification with the imbalanced class distribution problem, accuracy is no longer an appropriate measure since the minority classes have a very little impact on the accuracy compared to the majority classes [37]. Therefore, in this study, the F1-score is used to evaluate the models because the F1-score ( $2 \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$ ) shows an insight into the balance between sensitivity (recall) ( $\frac{TP}{TP+FN}$ ) and precision ( $\frac{TP}{TP+FP}$ ). This metric is also widely used in activity recognition [15, 21]

## Results and Discussion

In this section, the results of the experiments using LSTM and CNN are presented and discussed in the aspect of different methods of handling imbalanced classes and different feature extraction approaches. FTWs and ESTWs are used to pre-process data and build the datasets for training. SMOTE, cost-sensitive and ensemble learning methods are used for handling the class imbalance present in the datasets. Table 3 shows the results of the F1-score of the LSTM and CNN models from the home A for the imbalanced dataset, with cost-sensitive corrections and minority sampling

**Table 3** F1-score Home A

| Activity       | FTWs            |             |               |             |             |             |             | ESTWs           |             |               |             |             |             |             |
|----------------|-----------------|-------------|---------------|-------------|-------------|-------------|-------------|-----------------|-------------|---------------|-------------|-------------|-------------|-------------|
|                | Imbalanced data |             | Cos-Sensitive |             | SMOTE       |             | Ensemble    | Imbalanced Data |             | Cos-Sensitive |             | SMOTE       |             | Ensemble    |
|                | CNN             | LSTM        | CNN           | LSTM        | CNN         | LSTM        |             | CNN             | LSTM        | CNN           | LSTM        | CNN         | LSTM        |             |
| Snack          | 0.00            | 0.00        | 0.00          | 0.00        | 0.28        | 0.39        | 0.00        | 0.00            | 0.00        | 0.00          | 0.00        | 0.27        | 0.42        | 0.01        |
| Showering      | 0.36            | 0.48        | 0.43          | 0.47        | 0.70        | 0.70        | 0.51        | 0.79            | 0.81        | 0.82          | 0.81        | 0.89        | 0.89        | 0.82        |
| Grooming       | 0.00            | 0.00        | 0.00          | 0.00        | 0.25        | 0.28        | 0.12        | 0.55            | 0.53        | 0.54          | 0.55        | 0.56        | 0.57        | 0.57        |
| Breakfast      | 0.61            | 0.67        | 0.65          | 0.68        | 0.71        | 0.73        | 0.38        | 0.71            | 0.72        | 0.76          | 0.74        | 0.73        | 0.77        | 0.67        |
| Toileting      | 0.00            | 0.00        | 0.00          | 0.00        | 0.31        | 0.37        | 0.17        | 0.00            | 0.00        | 0.00          | 0.00        | 0.28        | 0.29        | 0.17        |
| Lunch          | 0.75            | 0.80        | 0.81          | 0.82        | 0.80        | 0.84        | 0.64        | 0.81            | 0.80        | 0.82          | 0.85        | 0.86        | 0.86        | 0.81        |
| Leaving        | 0.76            | 0.86        | 0.75          | 0.83        | 0.88        | 0.89        | 0.83        | 0.85            | 0.86        | 0.86          | 0.86        | 0.87        | 0.87        | 0.84        |
| Sleeping       | 0.96            | 0.96        | 0.96          | 0.96        | 0.92        | 0.90        | 0.92        | 0.99            | 0.99        | 0.99          | 0.99        | 0.99        | 0.99        | 0.99        |
| Spare Time     | 0.91            | 0.91        | 0.90          | 0.91        | 0.92        | 0.93        | 0.76        | 0.98            | 0.98        | 0.98          | 0.98        | 0.99        | 0.99        | 0.98        |
| <b>Average</b> | <b>0.44</b>     | <b>0.48</b> | <b>0.46</b>   | <b>0.47</b> | <b>0.63</b> | <b>0.67</b> | <b>0.48</b> | <b>0.60</b>     | <b>0.62</b> | <b>0.62</b>   | <b>0.63</b> | <b>0.71</b> | <b>0.73</b> | <b>0.65</b> |

using SMOTE. The F1-score of the minority classes which are *Breakfast*, *Grooming*, *Lunch*, *Showering*, *Toileting* and *Snack* from the home A are improved using SMOTE based on both approaches of extracting features and both models. The results also show the majority classes which are *Leaving* and *Spare-Time* activities (except *Sleeping*) which are also improved based on both approaches of extracting features for both models using the SMOTE method. The average results of the LSTM and CNN for all activities are improved using the SMOTE method based on both FTWs and ESTWs. Regarding home B, the F1-score of the minority classes (*Breakfast*, *Grooming*, *Lunch*, *Showering*, *Toileting*, *Snack*, and *Dinner*) is considerably improved, which are shown in Table 4. Moreover, only the results of the *Spare-Time* as the majority classes are improved based on FTWs. The average results of home B indicate that the SMOTE method substantially improved the recognition, particularly for the minority classes. The F1-scores in Tables 3 and 4 indicate that the results of the models based on both feature extraction approaches using SMOTE are better (higher F1-score) than the results of models based on cost-sensitive and class imbalanced datasets. Moreover, the F1-score results based on SMOTE with ESTWs can be seen to be higher than F1-scores based on SMOTE with FTWs from both homes of both models on average. Moreover, the obtained results based on the SMOTE technique with both feature extraction method (FTW and ESTW) and with both temporal models (LSTM and CNN) are better than the results obtained by balanced ensemble learning as shown in Tables 3 and 4. Therefore, the proposed data-level solution (SMOTE and ESTWs) to handle imbalanced human activities from smart homes is more promising than algorithms level (cost-sensitive and ensemble learning).

## Conclusion and Future Work

Human activity recognition is a dynamic and challenging research area that plays an important role in diverse applications such as smart environments, security, health care, elderly care, emergencies, surveillance and context-aware systems. The frequency and duration of human activities are intrinsically imbalanced. The huge difference in the number of observations for the classes to learn will make many machine learning algorithms to focus on the classification of the majority examples due to its increased prior probability while ignoring or misclassifying minority examples. In this study, SMOTE and cost-sensitive learning are applied to temporal models and compared with ensemble learning to handle the class imbalance problem as well as to study the relation to two data pre-processing methods. Experiments show that f-measures of the minority classes are increased when using SMOTE with both temporal models (LSTM and CNN) and based on both ways of extracting features (FTWs and ESTWs). For example, the recognition measurement of the *Snack* and *Dinner* as one of the minority classes is notably improved in both homes, using both models and based on both feature extraction methods. The experimental results indicate that handling imbalanced data is more important than selecting machine learning algorithms and improves classification performance. Moreover, handling imbalanced class problem from data level using SMOTE and ESTWs for these activity datasets outperforms the algorithm level.

Future work will explore a newly proposed approach to handle the imbalanced class problem by integrating SMOTE with weak supervision. This approach will use SMOTE only to generate observations from minority classes and use weak supervision to correctly and properly label the new

**Table 4** F1-score Home B

| Activity       | FTWs            |             |               |             |             |             |             | ESTWs           |             |               |             |             |             |             |
|----------------|-----------------|-------------|---------------|-------------|-------------|-------------|-------------|-----------------|-------------|---------------|-------------|-------------|-------------|-------------|
|                | Imbalanced data |             | Cos-Sensitive |             | SMOTE       |             | Ensemble    | Imbalanced data |             | Cos-Sensitive |             | SMOTE       |             | Ensemble    |
|                | CNN             | LSTM        | CNN           | LSTM        | CNN         | LSTM        |             | CNN             | LSTM        | CNN           | LSTM        | CNN         | LSTM        |             |
| Dinner         | 0.00            | 0.00        | 0.00          | 0.00        | 0.31        | 0.34        | 0.06        | 0.00            | 0.01        | 0.00          | 0.00        | 0.26        | 0.27        | 0.13        |
| Snack          | 0.00            | 0.00        | 0.02          | 0.08        | 0.27        | 0.29        | 0.22        | 0.00            | 0.00        | 0.00          | 0.00        | 0.26        | 0.28        | 0.07        |
| Showering      | 0.00            | 0.22        | 0.00          | 0.21        | 0.26        | 0.36        | 0.24        | 0.73            | 0.80        | 0.71          | 0.79        | 0.82        | 0.84        | 0.53        |
| Grooming       | 0.13            | 0.30        | 0.09          | 0.30        | 0.39        | 0.36        | 0.42        | 0.62            | 0.61        | 0.61          | 0.61        | 0.64        | 0.65        | 0.54        |
| Breakfast      | 0.50            | 0.47        | 0.51          | 0.51        | 0.52        | 0.58        | 0.36        | 0.26            | 0.23        | 0.24          | 0.19        | 0.30        | 0.35        | 0.29        |
| Toileting      | 0.00            | 0.00        | 0.00          | 0.00        | 0.31        | 0.32        | 0.32        | 0.23            | 0.04        | 0.23          | 0.10        | 0.26        | 0.27        | 0.14        |
| Lunch          | 0.39            | 0.35        | 0.31          | 0.38        | 0.41        | 0.42        | 0.37        | 0.00            | 0.00        | 0.00          | 0.00        | 0.36        | 0.38        | 0.00        |
| Leaving        | 0.90            | 0.90        | 0.89          | 0.89        | 0.90        | 0.90        | 0.84        | 0.66            | 0.66        | 0.66          | 0.66        | 0.66        | 0.66        | 0.66        |
| Sleeping       | 0.97            | 0.97        | 0.97          | 0.97        | 0.97        | 0.97        | 0.97        | 0.97            | 0.97        | 0.97          | 0.97        | 0.97        | 0.97        | 0.97        |
| Spare Time     | 0.83            | 0.82        | 0.84          | 0.84        | 0.85        | 0.86        | 0.79        | 0.90            | 0.90        | 0.90          | 0.90        | 0.89        | 0.90        | 0.90        |
| <b>Average</b> | <b>0.33</b>     | <b>0.36</b> | <b>0.36</b>   | <b>0.41</b> | <b>0.51</b> | <b>0.54</b> | <b>0.45</b> | <b>0.40</b>     | <b>0.40</b> | <b>0.40</b>   | <b>0.40</b> | <b>0.54</b> | <b>0.56</b> | <b>0.42</b> |



observations. The idea is designed to target the challenge of correctly labeling samples created in an over-sampling context. The long-term goal of our project will work on boosting learning across different smart homes aiming to perform robust recognition of dangerous situations and detect behavior deviations in order to enhance elderly care alert systems. This will be conducted by transferring knowledge over different smart homes in terms of layout, resident and sensor configuration.

**Acknowledgements** Open access funding provided by Halmstad University. This research is supported by the Knowledge Foundation under the project of the Center for Applied Intelligent Systems, under Grant Agreement No. 20100271.

## Compliance with ethical standards

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

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. What is healthy ageing? <https://www.who.int/ageing/healthy-ageing/en/>. Accessed: 2019-08-10.
2. Ali Hamad Rebeen, Järpe Eric, Lundström Jens. Stability analysis of the t-sne algorithm for human activity pattern data. In *The 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC2018)*, 2018.
3. Bai Shaojie, Zico Kolter J, Koltun Vladlen. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint [arXiv:1803.01271](https://arxiv.org/abs/1803.01271), 2018.
4. Banos Oresti, Galvez Juan-Manuel, Damas Miguel, Pomares Hector, Rojas Ignacio. Window size impact in human activity recognition. *Sensors*. 2014;14(4):6474–99.
5. Cao Liang, Wang Yufeng, Zhang Bo, Jin Qun, V Vasilakos Athanasios. Gchar: An efficient group-based context-aware human activity recognition on smartphone. *Journal of Parallel and Distributed Computing*. 2018;118:67–80.
6. Manosha Chathuramali KG, Rodrigo Ranga. Faster human activity recognition with svm. In *Advances in ICT for Emerging Regions (ICTer), 2012 International Conference on*, pages 197–203. IEEE, 2012.
7. Chawla Nitesh V, Bowyer Kevin W, Hall Lawrence O, Philip Kegelmeyer W. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*. 2002;16:321–57.
8. Collins Jasmine, Sohl-Dickstein Jascha, Sussillo David. Capacity and trainability in recurrent neural networks. *stat*. 2017;28:1050.
9. Das Barnan, Seelye Adriana M, Thomas Brian L, Cook Diane J, Holder Larry B, Schmitter-Edgemore Maureen. Using smart phones for context-aware prompting in smart environments. In *2012 IEEE Consumer Communications and Networking Conference (CCNC)*, pages 399–403. IEEE, 2012.
10. Devarakonda Aditya, Naumov Maxim, Garland Michael. Adabatch: Adaptive batch sizes for training deep neural networks. arXiv preprint [arXiv:1712.02029](https://arxiv.org/abs/1712.02029), 2017.
11. Espinilla M, Medina J, Hallberg J, Nugent C. A new approach based on temporal sub-windows for online sensor-based activity recognition. *J Ambient Intell Human Comput*. 2018. <https://doi.org/10.1007/s12652-018-0746-y>.
12. Fatima Iram, Fahim Muhammad, Lee Young-Koo, Lee Sung-oung. Analysis and effects of smart home dataset characteristics for daily life activity recognition. *The Journal of Supercomputing*. 2013;66(2):760–80.
13. Fung Gabriel Pui Cheong, Yu Jeffrey Xu, Wang Haixun, Cheung David W, Liu Huan. A balanced ensemble approach to weighting classifiers for text classification. In *Sixth International Conference on Data Mining (ICDM'06)*, pages 869–873. IEEE, 2006.
14. Galar Mikel, Fernandez Alberto, Barrenechea Ederne, Bustince Humberto, Herrera Francisco. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*. 2011;42(4):463–84.
15. Hamad R. A, Salguero A. G, Bouguelia M, Espinilla M, Quero J. M. Efficient activity recognition in smart homes using delayed fuzzy temporal windows on binary sensors. *IEEE Journal of Biomedical and Health Informatics*, pages 1–1, 2019. ISSN 2168-2194. <https://doi.org/10.1109/JBHI.2019.2918412>.
16. Hammerla Nils Y, Halloran Shane, Ploetz Thomas. Deep, convolutional, and recurrent models for human activity recognition using wearables. arXiv preprint [arXiv:1604.08880](https://arxiv.org/abs/1604.08880), 2016.
17. Huang Chen, Li Yining, Change Loy Chen, Tang Xiaou. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5375–5384, 2016.
18. Japkowicz Nathalie, Stephen Shaju. The class imbalance problem: A systematic study. *Intelligent data analysis*. 2002;6(5):429–49.
19. Jing Luyang, Wang Taiyong, Zhao Ming, Wang Peng. An adaptive multi-sensor data fusion method based on deep convolutional neural networks for fault diagnosis of planetary gearbox. *Sensors*. 2017;17(2):414.
20. Johnson Justin M, Khoshgoftaar Taghi M. Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1):27, Mar 2019. ISSN 2196-1115. <https://doi.org/10.1186/s40537-019-0192-5>.
21. Kasteren TL, Englebienne Gwenn, Kröse BJ. An activity monitoring system for elderly care using generative and discriminative models. *Personal and ubiquitous computing*. 2010;14(6):489–98.
22. Khan Salman H, Hayat Munawar, Bennamoun Mohammed, Sohel Ferdous A, Togneri Roberto. Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE transactions on neural networks and learning systems*. 2017;29(8):3573–87.
23. Lara Oscar D, Labrador Miguel A, et al. A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys and Tutorials*. 2013;15(3):1192–209.
24. Li Frédéric, Shirahama Kimiaki, Nisar Muhammad Adeel, Köping Lukas, Grzegorzec Marcin. Comparison of feature learning methods for human activity recognition using wearable sensors. *Sensors*. 2018;18(2):679.
25. Medina-Quero Javier, Orr Claire, Zang Shuai, Nugent Chris, Salguero Alberto, Espinilla Macarena. Real-time recognition of interleaved activities based on ensemble classifier of long short-term memory with fuzzy temporal windows. In *Multidisciplinary*

- Digital Publishing Institute Proceedings*, volume 2, page 1225, 2018a.
26. Medina-Quero Javier, Zhang Shuai, Nugent Chris, Espinilla M. Ensemble classifier of long short-term memory with fuzzy temporal windows on binary sensors for activity recognition. *Expert Systems with Applications*. 2018b;114:441–53.
  27. Mokhtari G, Aminikhanghahi S, Zhang Qing, Cook Diane J. Fall detection in smart home environments using uwb sensors and unsupervised change detection. *Journal of Reliable Intelligent Environments*. 2018;4(3):131–9.
  28. Rueda Fernando Moya, Grzeszick René, Fink Gernot, Feldhorst Sascha, Hompel Michael ten. Convolutional neural networks for human activity recognition using body-worn sensors. In *Informatics*, volume 5, page 26. Multidisciplinary Digital Publishing Institute, 2018.
  29. Murad Abdulmajid, Pyun Jae-Young. Deep recurrent neural networks for human activity recognition. *Sensors*. 2017;17(11):2556.
  30. Nguyen Ky Trung, Portet Francois, Garbay Catherine. Dealing with Imbalanced data sets for Human Activity Recognition using Mobile Phone Sensors. In *3rd International Workshop on Smart Sensing Systems, June 2018, Rome, Italy*, 2018.
  31. Nweke HF, Teh YW, Al-Garadi MA, Alo UR. Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges. *Expert Syst Appl*. 2018;105:233–61.
  32. Ordóñez F, De Toledo P, Sanchis A, et al. Activity recognition using hybrid generative/discriminative models on home environments using binary sensors. *Sensors*. 2013;13(5):5460–77.
  33. Park Jiho, Jang Kiyoun, Yang Sung-Bong. Deep neural networks for activity recognition with multi-sensor data in a smart home. In *Internet of Things (WF-IoT), 2018 IEEE 4th World Forum on*, pages 155–160. IEEE, 2018.
  34. Singh Deepika, Merdivan Erinc, Hanke Sten, Kropf Johannes, Geist Matthieu, Holzinger Andreas. Convolutional and recurrent neural networks for activity recognition in smart environment. In *Towards integrative machine learning and knowledge extraction*, pages 194–205. Springer, 2017.
  35. Srivastava Nitish, Hinton Geoffrey, Krizhevsky Alex, Sutskever Ilya, Salakhutdinov Ruslan. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*. 2014;15(1):1929–58.
  36. Stikic Maja, Huynh Tâm, Van Laerhoven Kristof, Schiele Bernt. Adl recognition based on the combination of rfid and accelerometer sensing. In *Pervasive Computing Technologies for Healthcare, 2008. PervasiveHealth 2008. Second International Conference on*, pages 258–263. IEEE, 2008.
  37. Sun Yanmin, Kamel Mohamed S, Wong Andrew KC, Wang Yang. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition*. 2007;40(12):3358–78.
  38. Tapia Emmanuel Munguia, Intille Stephen S, Larson Kent. Activity recognition in the home using simple and ubiquitous sensors. In *International conference on pervasive computing*, pages 158–175. Springer, 2004.
  39. Wang Jindong, Chen Yiqiang, Hao Shuji, Peng Xiaohui, Lisha Hu. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*; 2018.
  40. Wu Qiong, Zeng Zhiwei, Lin Jun, Chen Yiqiang. Ai empowered context-aware smart system for medication adherence. *International Journal of Crowd Science*, 2017.
  41. Yahaya Salisu Wada, Lotfi Ahmad, Mahmud Mufti. A consensus novelty detection ensemble approach for anomaly detection in activities of daily living. *Applied Soft Computing*. 2019;83:105613.
  42. Yala Nawel, Fergani Belkacem, Fleury Anthony. Feature extraction for human activity recognition on streaming data. In *Innovations in Intelligent Systems and Applications (INISTA), 2015 International Symposium on*, pages 1–6. IEEE, 2015.
  43. Yang Jianbo, Nguyen Minh Nhut, San Phyo Phyo, Li Xiaoli, Krishnaswamy Shonali. Deep convolutional neural networks on multichannel time series for human activity recognition. In *Ijcai*, volume 15, pages 3995–4001, 2015.
  44. Zhen LIU, Qiong LIU. Studying cost-sensitive learning for multi-class imbalance in internet traffic classification. *The Journal of China Universities of Posts and Telecommunications*. 2012;19(6):63–72.
  45. Zhou Zhi-Hua, Liu Xu-Ying. On multi-class cost-sensitive learning. *Computational Intelligence*. 2010;26(3):232–57.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.