Personal and Ubiquitous Computing
https://doi.org/10.1007/s00779-018-01196-8

ORIGINAL ARTICLE

Fall detection system for elderly people using IoT and ensemble machine learning algorithm

Diana Yacchirema[1,2] · Jara Suárez de Puga[2] · Carlos Palau[2] · Manuel Esteve[2]

Received: 22 August 2018 / Accepted: 14 December 2018

Abstract

Falls represent a major public health risk worldwide for the elderly people. A fall not assisted in time can cause functional impairment in an elderly and a significant decrease in his mobility, independence, and life quality. In this sense, we propose IoTE-Fall system, an intelligent system for detecting falls of elderly people in indoor environments that takes advantages of the Internet of Thing and the ensemble machine learning algorithm. IoTE-Fall system employs a 3D-axis accelerometer embedded into a 6LowPAN wearable device capable of capturing in real time the data of the movements of elderly volunteers. To provide high efficiency in fall detection, in this paper, four machine learning algorithms (classifiers): decision trees, ensemble, logistic regression, and Deepnets are evaluated in terms of AUC ROC, training time and testing time. The acceleration readings are processed and analyzed at the edge of the network using an ensemble-based predictor model that is identified as the most suitable predictor for fall detection. The experiment results from collection data, interoperability services, data processing, data analysis, alert emergency service, and cloud services show that our system achieves accuracy, precision, sensitivity, and specificity above 94%.

Keywords Fall detection · Internet of Things · 6LowPAN · IoT gateway · Ensemble learning algorithm · Random Forest · Accelerometer sensor · Elderly people

1 Introduction and related work on fall detection

Life expectancy has increased by a rate of 5 years since 2000 due to advances in the medical field. According to the World Health Organization (WHO), by 2050, the current population of elderly people (8.5%) will increase, representing 20% of the world's population [1]. On the basis of these trends, many countries are adopting healthy aging policies with the aim of helping elderly people lead an active and independent life [2]. In particular, ensuring an active and healthy aging (AHA) of the elderly people is one of the greatest challenges, but also a great opportunity for society in the coming decades. The notion of AHA has been lately characterized as a broad concept, which seeks to improve the quality of life (QoL) of the elderly people as they age, optimizing opportunities for health, participation, and security [2]. In that sense, health problems of the elderly people have become increasingly urgent [3] and falls are the most common accidents occurred whose severity may often require medical attention. According to the WHO, approximately 30% of the people over 65 suffer accidentally one or more falls per year, and for the people over 80 years, this rate increases to reach 50%. This figure is more alarming if one considers that falls often happen in indoor environments and are related to normal activities of daily living (ADL).

1.1 Problem area and methods

A serious consequence of suffering a fall is the "long lie," which consists of remaining on the ground for long periods of time until help arrives [4]. The long lie can lead to serious health complications, including dehydration, pneumonia, and hypothermia, which in many cases can lead to death within 6 months after a fall. Therefore, a fall not assisted in time in an elderly person can negatively impact their QoL and independence [4]. In this context, the development of real-time IoT

✉ Diana Yacchirema
diayac1@doctor.upv.es

[1] Escuela Politécnica Nacional, Ladrón de Guevara E11-253, 17-01-2759 Quito, Ecuador

[2] Universitat Politècnica de València, Camino de Vera S/N, 46022 Valencia, Spain

Published online: 02 January 2019

systems that contribute to efficiently detect falls and alert emergency services on the briefest possible time is a social need. Need that we aim to cover with the current work.

To overcome this problem we have followed a research methodology focused in the action design research [5] based, for one side, on the development of artifacts with the explicit intention of improving the functional performance of the artifact or solving an immediate problem, and for the other side, on the progression of problem solving led by individuals working to improve the way they address issues and solve problems.

Thus, the research methodology has been composed of the consecutive phases: (i) first, a dissection and definition of the problematic from a practical point of view and gathering of the requirements to solve it; (ii) second, a research into the literature and projects until the date that have been trying to solve this problematic, falls detection in elderly people, creating a state of the art (SotA) to be used as a background; (iii) third, together with the background study and requirements, a blueprint architecture has been defined in order to solve the problem in an optimal manner; (iv) also, having this blueprint, the fourth phase includes the study of technologies and protocols that better implements the architecture, and for that reason, this phase includes the research on machine learning algorithms and its testing to choose the algorithm that better fulfills the requirements of the proposal; (v) the next phase is the creation of a real prototype following the defined architecture and implemented with the chosen technologies; (vi) finally, the testing and performance indicators obtained from the solution, once the performance results are gathered, we proceed with its evaluation and comparison with the results presented in other projects. From the ones described in the SotA section, we consider those that presented quantitative performance results.

As this is also a progressive method, future iterations over the work will be done in order to improve the system and include more practical characteristics to the current solution.

1.2 Background and state-of-the-art analysis

At present, several solutions have been proposed for elderly fall detection. Such solutions are categorized [6] into three main types according to the sensor-technology used: non-wearable-based systems (NWS), wearable-based systems (WS), and fusion or hybrid-based systems (FS).

In particular, plenty of NWS systems that use vision-based devices such as [7–13] have been proven to be powerful and robust to detect falls. However, the main disadvantages of these systems are their high cost, in both equipment investment and computational resources for image processing, and the consequent lack of privacy for elderly people since these systems require cameras to be strategically distributed in the indoor environment in which they live.

To overcome these limitations, diverse WS systems have been proposed in the literature, which usually employ inertial sensors such as accelerometers and gyroscopes, typically attached to the body of the elderly for movement recognition when a fall takes place.

Some of these studies [14–16] take advantage of the smartphone built-in accelerometer, and gyroscope [17, 18], for continuously monitoring of the movement of elderly people. Additionally, other studies combine the use of smartphone sensors with external wearable sensors located in diverse body parts such as the wrist, the waist, the chest, the ankle [19], shoulder, and foot [20], to obtain a more complete dataset of information regarding the movement of the subjects. Also, the smartphone has been combined with other devices [21], as a smartwatch, to double-check the results of the fall detection algorithm from two different data entry points. Nevertheless, the continuous monitoring and collecting of data, in some cases together with the executing of fall detection algorithm applications in the smartphone, results in a relatively high energy consumption, allowing the system to be active just for short periods of time.

Thus, in order to fill gaps, wearable devices with accelerometers are being used increasingly in WS systems because they offer low power consumption, low cost, low weight, ease of operation, small size, the possibility of being mounted on various body locations and, most importantly, portability.

As a result, one of the most commonly used methods for fall detection involves the use of a tri-axial accelerometer along with a threshold-based algorithm, which has been used by some representative works [22, 23] [24, 25]. These works detect a fall when the acceleration coming from the tri-axial accelerometer embedded in a wearable device is out of the set threshold. One of the biggest advantages for using the threshold-based methods is less complexity and computation cost compared to other methods. However, finding an appropriate value for the threshold that allows detecting all type of falls without getting confused with some ADL has proved to be a complicated problem.

Recently, WS systems based on machine learning (ML) techniques have been proposed to deal with this problem and improve the accuracy in detecting falls. ML is a technique in computer science that involves statistical inference of models from data in order to make automated predictions. ML builds a model from training data to predict or solve the given problem [26].

In several works, the authors of WS focus on using some ML algorithms for dealing with fall detection and increasing accuracy. For example, the non-linear support vector machine (SVM) algorithm has been used by several works as [27–29] in order to extract the features and obtain meaning from the human body. Mezghani et al. [27] captured fall and orientation data by an accelerometer attached to a smart textile. In the same way, Pierleoni et al. [28] also detected the orientation of the subject wearing incorporated and combined information from 3-axis accelerometer, 3-axis gyroscope, and 3-axis magnetometer. Since these systems need two feature extractions: the first to identify the peak and the second to detect the fall orientation, it requires more processing compared to the algorithms carrying out an only extraction may not be supported in real-time applications with constrained processing capabilities.

Aziz et al. [29] found that SVM is the best fall classification accuracy by evaluating five different learning algorithms (logistic regression, Naïve Bayes, decision trees, K-nearest neighbor, and SVM) in terms of three measures of performance: sensitivity (SE), specificity (SP), and false-positive rate. The system was tested incorporating waist-mounted triaxial accelerometers of young volunteers. The SVM algorithm achieved the highest combination of SE (96%) and SP (96%) in distinguishing falls from ADLs. In another comparative work between SVM, k-nearest neighbor (k-NN), and multi-layer perceptron (MLP) machine learning algorithms, described by Nguyen et al. [30], the authors found that MPL evaluated on an open dataset was the algorithm that obtained better results in detecting falls, with an SP, SE, and accuracy of over 98%. However, the validation provided by SP and SE, accuracy may not be sufficient when the aim is to reduce the time of long lie. In addition, these solutions do not alert when a fall event occurs.

A similar study carried out by Özdemir et al. [31] also compared the performance and the computation complexity achieved by six machine learning algorithms in the distinction of falls from ADLs. The k-NN, least squares method (LSM), support vector machines (SVMs), Bayesian decision-making (BDM), dynamic time warping (DTW), and artificial neural network (ANN) algorithms were evaluated in terms of SE, SP, accuracy, time training, and testing time. The authors used the data from three tri-axial sensors (accelerometer, gyroscope, and magnetometer) placed at six different positions of participating volunteers for evaluating algorithm performance. In this case, they found that K-NN and LSM algorithm provided SE, SP, and accuracy above 99%. Attending at the computational time in both, training and testing phases, LSM provided promising results compared to k-NN, obtaining a minimum of 2.2 ms of delay in the training phase and up to 32.7 ms at testing. While k-NN provided 318.2 and 76.6 ms in the training and testing phase, respectively.

Santoyo et al. [19] performed a systematic assessment study between SVM, K-NN, decision trees, and Native Bayes algorithm, with four sensors located in different body positions as chest, waist, ankle, and thigh for discriminating among falls and ADLs using on analysis of variance (ANOVA). On one side, the authors demonstrated that the chest and waist are the two most suitable position to locate the sensors and maximize the system effectiveness, with waist position as the more ergonomic alternative. On the other hand, the authors found that the SVM algorithm with a sensor placed on the waist achieved an SE and SP higher to 93%. However, the processing data was carried out on the smartphone, and it may be a point of failure in the system due to the inherent limitation of the battery. In addition, this solution does not send any alert messages when a fall is identified.

By extracting time series from human motion retrieved by a tri-axial accelerometer placed at human upper trunk, Tong et al. [32] used the hidden Markov model (HMM)-based method to detect and predict falls. The experiment results show an ideal success rate in the fall detection (100% sensitivity and 100% specificity). However, this solution only uses samples of young people's simulated activities for training and setting the HMM (λ) and thresholds of the system and it does not send alert notifications when a fall takes place.

Other study proposed by Aguiar et al. [16] used a smartphone built-in accelerometer for continuously monitoring the movement's data of elderly people. These data were used to test offline three different learning classifiers: decision trees, K-nearest neighbors (k-NN), and Naive Bayes. The results show that the decision tree-based algorithm presented the best performance, with more equilibrated sensitivity and specificity values compared with the other tested algorithms. Nevertheless, as a result of the relatively high energy consumption of smartphones, this system could only be active for a short period of time.

Finally, Wang et al. [12] performed evaluations to distinguish falls from the other activities of elderly people in indoor environments with different Tx-Rx layout schemes by using existing wireless infrastructures and laptops equipped with 802.11n NIC through WiFall system. WiFall employed SVM and ensemble (random forest) algorithms to classify both ADLs and falls. The authors found that ensemble improves classification accuracy up to 6% (in one of the experimental scenarios) compared with SVM. Moreover, WiFall sends an alarm signal to the client when a fall takes place. However, these results reveal that false alarm is above 12%. In addition, the system was tested with only one single person.

In this work, we have followed a similar approach by using a wearable sensor located at the waist of elderly people and an ensemble (Random Forest) algorithm for fall detection, but we differ from previous works in the way the system is built. First, the data from the movements of elderly people in the indoor environment is captured by a 3D-axis accelerometer embedded into a 6LowPAN device wearable. Second, the fall of an elder is detected by an ensemble consisting of 10 decision trees which is built and training in the cloud but that are instantiated at the edge of the network where the detection phase takes place. Prior to this, a dataset, Motion-DT of falls, and activities of daily living were built from the historical knowledge of a publicly accessible dataset that contains records of usual falls and ADL of elderly people using the sliding-windows technique along with the normalized signal magnitude area (SMA). Subsequently, four machine learning algorithms (classifiers): decision trees, ensemble, logistic regression, and Deepnets are assessed by computational requirements (i.e., training time and testing time) and optimization parameters (i.e., ROC and AUC ROC), in order to select the most suitable learning classification algorithm to fall recognition using k-fold cross-validation. Third, the effectiveness of the proposed system to distinguish ADLs and falls detected is verified by experiments performed by adult volunteers in terms of accuracy, precision, sensitivity, and specificity.

Finally, one of the main innovations of the proposed system is that the process of fall detection is carried out on an IoT gateway which provides edge computing capabilities that enable (i) to reduce the time of long lie by detecting fall closer to where data were created, instead of sending all the data through long routes to cloud for processing; (ii) to send notifications with information of the fall type and location of elderly person's house to healthcare professionals through a lightweight and secure IoT protocol that also provides quality of service (QoS) levels; and (iii) the data transformation in uniform data format and interoperability of the system with ambient assisting living platforms.

Following Sect. 1, this paper is organized as follows. In Sect. 2, the IoTE-Fall-system stages and architecture are described in detail. In Sect. 3, the experimental results and evaluations are presented. Lastly, conclusions and future work are described in Sect. 4.

2 Fall detection system using IoT and ensemble machine learning algorithm

The block diagram depicted in Fig. 1 defines the fall detection system stages identified in this research: feature extraction, training and testing, and validation. The sliding-windows and SMA techniques are used to extract the features that describe the raw signal of the elderly person's movements from a publicly accessible dataset. These features are stored in a new dataset, Motion-DT, which is used in the k-fold cross-validation scheme to train and test four machine learning algorithms in order to find the best model for fall detection. The selected model is then validated with the acceleration measurements gathered from a MEMs accelerometer sensor with the purpose of obtaining the real performance of the fall detection system.

2.1 Features extraction

A suitable approach to training and testing fall detection systems is to use the historical knowledge coming from public datasets which consist of different past events (real or simulated falls, ADLs, or both) performed by a set of participants wearing different sensors located on various parts of the body. Historical knowledge is essential to understand what behavior is expected. For example, using knowledge of the behavior of unexpected motion patterns that have occurred when an adult fall will enable alert and predict situations of risk when similar patterns of behavior occur. However, the historical knowledge must be previously processed to extract the features that represent the raw signal as possible.

2.1.1 Historical knowledge from SisFall

In this section, we focus on identifying the periods of user activity on a publicly accessible dataset, SisFall from Sucerquia et al. [33] in order to extract the most representative features to make the prediction. The SisFall dataset contains records from 38 participants, of which 15 are elderly persons aged between 60 and 75 years old. All elderly participant simulated activities of ADLs and only one of these participants, who is an expert in Judo, simulated both, falls and ADLs. The signals were captured in laboratory experiments using three inertial sensors (2 accelerometers: ADXL345 and ITG3200, and 1 gyroscope: MMA8451Q) integrated into a self-developed embedded device. Both acceleration and gyroscope signals were acquired with a sampling rate of around 200 Hz. The elderly participant performed 19 types of ADLs and 15 types of fall. More detailed explanation concerning the

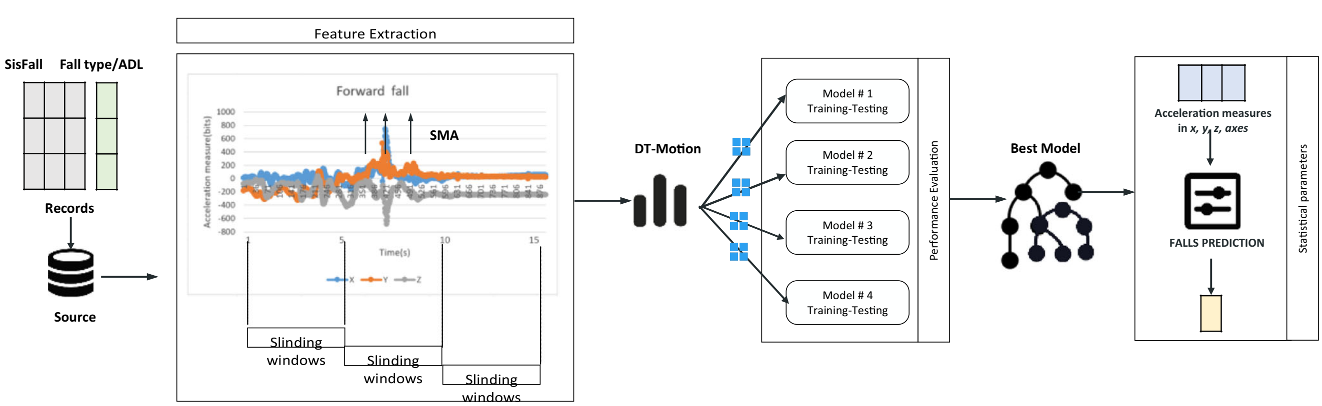

Fig. 1 System stages involved in the fall detection

SisFall protocol can be found in [33]. Several studies founded that there is a potentially clinically important variation in the type of fall that an elderly person can suffer [34]. In this study, we focus on the most common types of fall that can cause serious health risks to the elderly people. According to a study by Youn et al. [35], the falls more severe in elderly people occur in a forward direction when people are walking. On the other hand, fall in a backward direction is the most recurrent fall and it can cause serious damage to human health, especially like a fracture in the wrist and vertebral fracture. In additional, lateral fall is also dangerous because the elderly people can suffer a hip fracture [36]. Therefore, the falls included in this research were forward fall (FF), backward fall (BF), and lateral fall (LF). Moreover, three ADLs also were included in this research, namely walking (WA), stairs climbing (SCA), and sitting (SA), in order to distinguish falls from these fall-like activities. The trials of SisFall dataset regarding these three types of falls and ADLs are shown in Table 1.

2.1.2 Sliding windows along with SMA

The sliding-windows technique along with SMA was applied on SisFall trials to get the features that represent the changes of acceleration (peaks) produced by the falls and ADLs. SMA has been reported by previous authors [37] [38] as a suitable measurement for distinguishing between duty periods and rest periods using accelerometer data.

First, the sliding window was applied as the splitter technique of the SisFall data streams in order to locate the segment where the acceleration signal in all three axes suffer the highest variation. Owing to the fact that human activities and falls happen over a relatively short period of time, 5 s is a proper length window for human fall and ADL detection. This fixed-length window used to slide through the whole sequence is coherent with other fall detection-related researches. For instance, the authors in [39–41] employed the sliding windows 0.4 s, 0.6 s, and 0.5 s, respectively. In fact, according to a study by Lombardi [42], a length window of about 0.5 s seems to be the best compromise among effectiveness, complexity, and low power consumption to check the acceleration values within a fall event.

Table 1 Falls and activities of SisFall used in this work

Code	Description	Trials	Duration
FF	Forward fall	5	15 s
BF	Backward fall	5	15 s
LF	Lateral fall	5	15 s
WA	Walking	5	15 s
SCA	Stairs climbing	5	15 s
SA	Sitting	5	15 s

Subsequently, for each sample windows, the SMA calculation was performed according to (Eq. 1)

$$\mathrm{SMA} = \frac{1}{w}\sum_{i=1}^{w}\left(|x_i| + |y_i| + |z_i|\right) \tag{1}$$

where w is the window length; x_i,y_i, and z_i refer to the acceleration signal of each axis. The SMA is computable by summing of acceleration magnitude summations over three axes over 5 s.

Finally, the Motion-DT is obtained as a result of this process, which was carried out by running a program developed using the MatLab code. Motion-DT stores the signals with high variation along with the typology of the different falls and ADLs. In our system, the TD-Motion includes 1464 ADLs records and 1920 falls records.

2.2 Machine learning algorithm (classifier) selection

Using Motion-DT, we applied the k-fold cross-validation technique in order to select the most suitable supervised learning classification algorithm. In this study, a total of four different classification algorithms are selected and compared including decision trees, ensemble, logistic regression, and Deepnet. These classification algorithms are briefly introduced in the following section.

2.2.1 Machine learning algorithms

Logistic regression It is one of the machine learning algorithms that is being used increasingly in biomedicine. Also, it has been used to develop fall predictive models as in [43–45]. This algorithm computes, for each target class, a probability modeled as a logistic function value, whose parameters are a linear combination of its inputs. In particular, input fields ($X1, X2, \ldots. Xk$) are combined linearly using logistic regression coefficients ($b_{0,\ i}, b_{1,\ \ i}, b_{2,\ \ i}, \ldots. b_{k,\ \ i}$) to predict an output field (y), which is the probability (Eq. 2) for each of the i classes of the objective field:

$$pi = +\frac{1}{1+e^{-fi(x)}} \tag{2}$$

where

$$fi(X) = \left(b_{0,i} + b_{1,i} + b_{1,i}X_1 + b_{2,i}X_2 \ldots. + b_{k,i}X_k\right)$$

The logistic regression tries to learn the k coefficients of the linear function, $fi(X)$, using maximum likelihood estimation techniques. In particular, the shape of the curve that is used is a natural fit for dividing data into groups. Logistic regression works better in cases where the features are roughly linear and the problem can be linearly solved. This is mainly due to the fact that the logistic regression generates linear decision boundaries to separate the objective classes.

Deepnets This machine learning algorithm also has been successfully applied to fall detection in works such as [46] [47] [48] [49] [50]. Deepnets is composed of a set of interconnected layers in which the input values give rise to the output values through a series of nodes. Each node is essentially a function that transforms the input features into a collection of values with their corresponding weights. Between the input layer and the output layer, there may be one or several hidden layers.

Decision trees This algorithm is widely used for data classification and their effectiveness has been proven at addressing many problems, including fall detection, in works such as [16, 19, 43, 51, 52]. Decision trees are composed of nodes and branches that create a model of decisions with a tree graph using the divide-and-conquer algorithm. In other words, the decision tree model consists of a series of paths (branch nodes) that lead to decisions based on a prediction (leaf nodes).

Ensemble (random forest) Ensemble algorithm has achieved excellent results and has been proven to be very effective in fall detection [12, 53–55]. Random forest (RF) solves a classification problem through an ensemble of decision trees trained with a bagging mechanism. Decision trees are combined to create a strengthened model with better predictive performance than each of the individual decision trees. This is due to that an ensemble model is less sensitive toward outliers in training data. By learning multiple models over samples of data and taking a majority vote in the prediction time, the ensemble-RF models correct decision trees' habit of overfitting to their training and generalizes better when applied to new data by averaging away the errors of each individual decision tree [56]. With the appropriate configuration, it is one of the methods that, with less complexity, provide better results in a wide variety of machine learning problems because it utilizes the strengths of the individual decision trees, while at the same time, the decision tree weaknesses are circumvented [57].

2.2.2 Classifier selection

The selection of the suitable classification algorithm consisted of evaluating the performance of fall and ADL detection of the four afore-described classification algorithms. To this end, we applied the 5-fold cross-validation on Motion-DT using trial-based partitioning, i.e., we created five models for each classifier and train them on four trials (of each type of fall and ADL) and tested them on the remaining trials as depicted in the Fig. 2. We created the models on a private cloud server using the REST API of BigML data analysis tool [58]. BigML is a software-as-a-service to machine learning, designed to create predictive models and embed them into software applications by RESTful APIs. A FUJITSU server equipped with 64GB of memory and an Intel Xeon E3-1220v5 CPU running at 3.00 GHz is used in the cloud for creating and evaluating the models.

The performance of each classification algorithm is evaluated through the area beneath the ROC curve, also known as AUC, the training time, and the testing time. The training and testing times are identified as the length of time taken by a classification algorithm to build and test the model, respectively. ROC curve represents the existing trade-off between the sensitivity (SE) and specificity (SP) in a learning algorithm. The SE represents the true-positive rate (TPR) while that the SP represents the false-positive rate (FPR). Larger AUC generally implies better performance.

Since the correlation between ROC curves is not clearly dominating [59]. In this work, we used AUC to compare classifiers. The idea underlying the usage of the AUC is its ability to represent the expected performance through a single scalar value. The AUC has proven to be statistically consistent and more discriminating measure than accuracy, especially in the algorithm optimization [59]. This metric has been widely employed to evaluate machine learning algorithms in work, such as [60] [61] [53].

To get the estimation of each model's performance, ROC curves are plotted and AUC values are calculated for each validation based on [62]. In the same way, the training and testing times are computed. Then, the result of the 5

Fig. 2. 5-fold cross-validation applied to Motion-DT to select the best suitable classifier for fall detection

evaluations is averaged to obtain the final cross-validation measures. The AUC performance for all classifiers exhibits encouraging results for all types of falls and ADLs after the application of the cross-validation technique, as shown in Fig. 3. In addition, we observe that all classifiers present a better predictive performance for the FF (> 0.99) and WA (0.97) type of fall and ADL, respectively.

Table 2 shows the experimental results, which reveals that the decision trees, ensemble (RF), and Deepnet classifiers achieve an average AUC value above 0.97, which is greater than the logistic regression and significantly better than random guessing (0.50). Nevertheless, extensive training and testing times are required for the Deepnet and logistic regression algorithms. According to the terms of the required training time, the algorithms can be ranked as decision tree, logistic regression, ensemble, and Deepnet, with very little difference between the first three algorithms. The ensemble classifier achieved the first highest AUC value as well as the lowest testing time among the classifiers. This lower testing time is due to the fact that training and testing of each of the individual decision trees is performed in parallel. As a consequent, the ensemble classification algorithm is identified as an efficient classifier for our system using acceleration data and is the one selected to be implemented in the architecture.

Table 2 The evaluation of the various classifier algorithms

Supervised learning classification algorithm	Average AUC	Training time (s)	Testing time (s)
Decision tree	0.9748	4.52	3.52
Ensemble	0.9951	5.75	3.48
Logistic regression	0.8487	5.4	4.80
Deepnet	0.9906	223.6	3.8

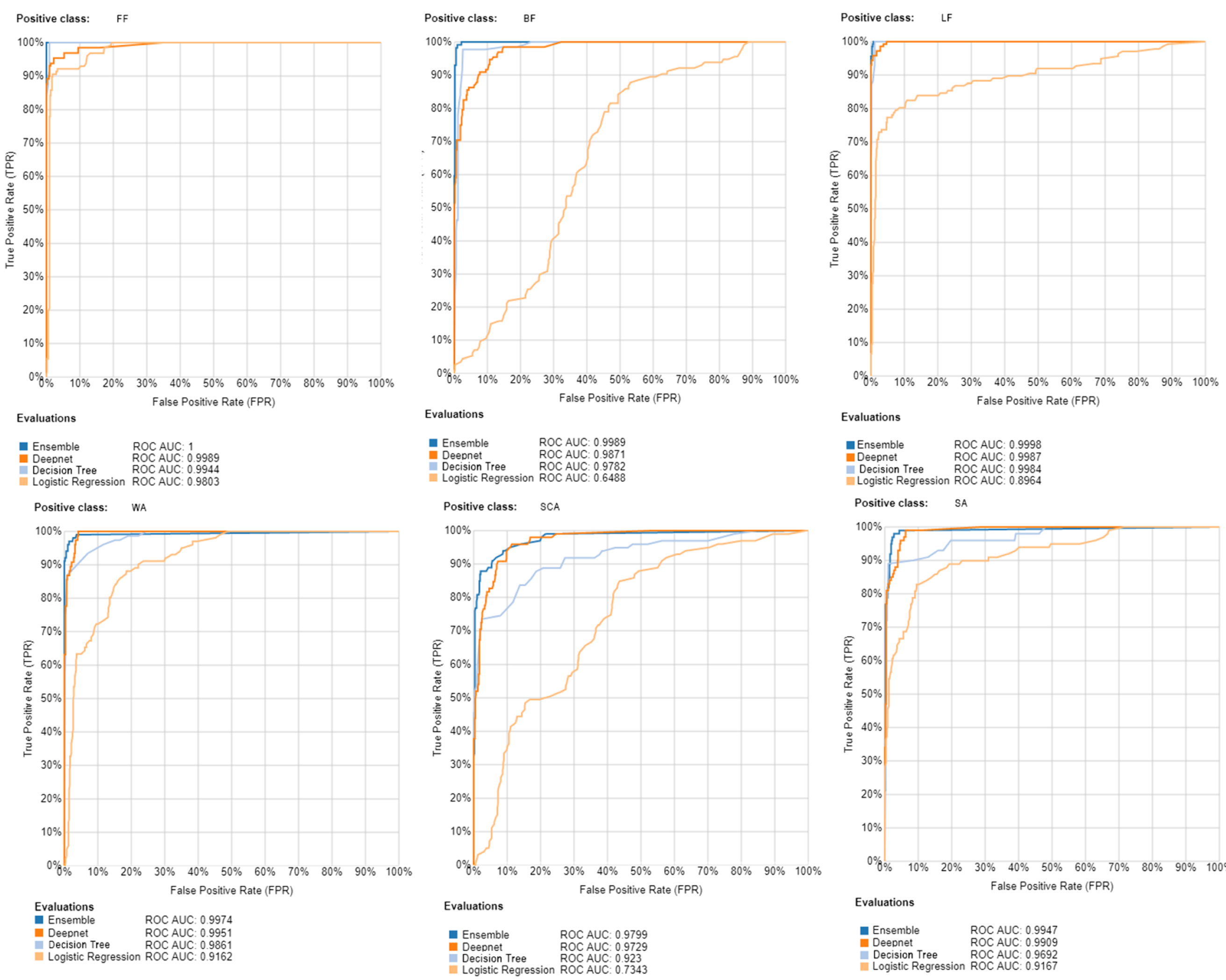

Fig. 3 Assessment of the different classifiers with fivefold cross-validation. The top panel shows the ROC curves and AUC values achieved for each type of falls with an average of 0.96. AUC. The bottom panel shows the ROC curves and AUC values achieved for each type of ADLs with an average of 0.95. AUC

Once the classification algorithm has been chosen, the classification model created from it is used in the fall detection system as described below.

2.3 IoTE-fall system

2.3.1 System overview

The proposed IoTE-Fall, shown in Fig. 4, consists of four main components: a wearable device, a wireless communication network, an IoT gateway, and Cloud services. Each component plays an important role in fall detection. The wearable device, interwoven with motion MEMs sensors, measures the acceleration (expressed in bits) of body movements of elderly people and transmits them to the IoT gateway using a low-power wireless area network. The IoT gateway processes and analyzes the received data (using ensemble-RF classifier) at the edge of the network to rapidly detect falls and act accordingly by sending alert messages in real time to the healthcare professionals concerned. The cloud is composed of services such as authentication, storage, and Big Data modeling.

2.3.2 Wearable device

A prototype of the wearable device was constructed from the combination of three key modular blocks: STM32 microcontroller, plugged with one expansion board based on the SPSGRF-915 with sub-1GHz RF connectivity operating at 868 or 915 MHz, and a motion MEMS sensor expansion board developed by ST Microelectronics. The STM32 microcontroller is equipped with an ARM 32-bit Cortex-M3 processor designed to offer very high performance, digital signal processing with low power, and low voltage operation. The sensor board consists of several tiny-ultralow-power sensors. However, only the MEMS motion sensor (LSM6DS0) is used for gathering the motion data that take place when the adult is falling or performing ADLs, and it can be worn comfortably without interfering with daily activities of the elderly. LSM6DS0 is a 3D-axis accelerometer which operates with a full-scale acceleration range ($\pm 2/\pm 4/\pm 8$ g). The schematic diagram of the detection device is shown in Fig. 5a, and the hardware structure is shown in Fig. 5b.

The wearable device firmware is based on "Contiki," an open-source operating system (OS) developed for constrained networks. By using the Contiki OS, we get full IoT stack support over 6LowPAN, in other words, 6lowPAN, RPL (IPv6 routing protocol for low-power and Lossy networks), and Constrained Application Protocol CoAP (through Erbium) support. Erbium is a low-power REST engine written in C language that provides RESTful access to the resources of the wearable device.

Furthermore, the REST paradigm, inherent in the constrained application protocol (CoAP), has been exploited. CoAP is a lightweight IoT protocol which shares similarities

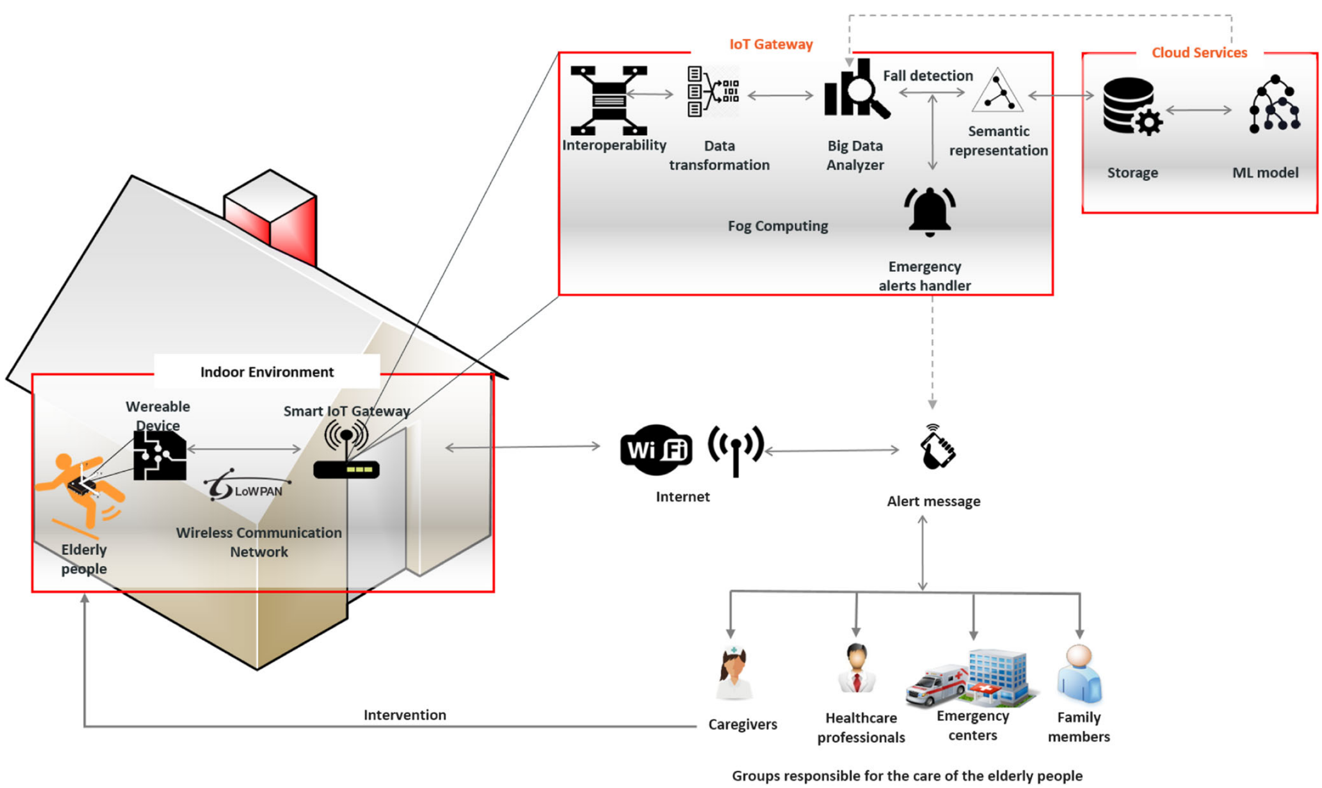

Fig. 4 Overview of the proposed IoTE-Fall system

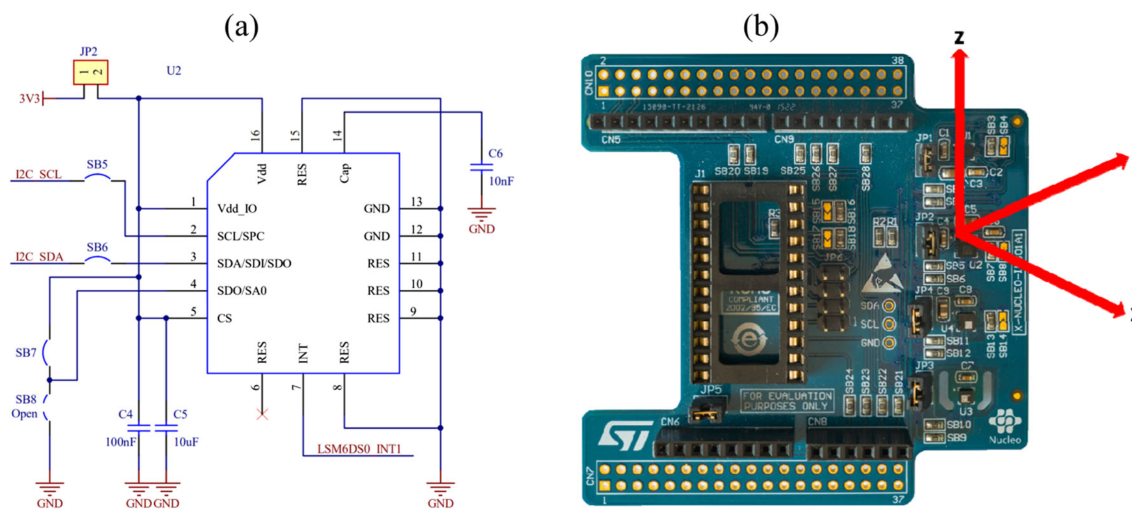

Fig. 5 Wearable device. **a** Block diagram schematic. **b** Hardware structure

with HTTP since it includes resource abstraction, URI use, RESTful interaction (i.e., methods such as GET, POST, PUT, and DELETE to access the various resources), and extensible header options [63]. However, compared with HTTP, CoAP implementation uses minimal resources on the constrained devices and constrained networks. Hence, it is suitable for constrained environments common in IoT deployments.

A CoAP server embedded in the wearable device is configured for reading the accelerometer measurements. The measurements are retrieved with a sampling rate of 100 Hz gathered using the CoAP GET method on the respective resource path including the IPv6 address and port of the CoAP server as shown in Table 3.

2.3.3 Wireless communication network

The wireless communication between devices and the Smart IoT gateway is established by the low-power wireless IPv6 (6LowPAN) technology based on the IEEE 802.15.4 standard. 6LoWPAN is a technology designed for supporting the connectivity, interoperability, and compatibility of heterogeneous wireless sensor networks (WSNs) at a very low cost and with very low requirements, compared with other technologies such as Wi-Fi or Bluetooth. In addition, this technology has inherent advantages as greater mobility, bigger address space, easy deployment, and maintaining, among others, what makes this technology suitable to be used in IoT-enabled devices, especially in resource-constrained devices.

Table 3 Acceleration resource and associated CoAP Path

Parameter	Description
Sensor	LSM6DS0
Resource	Acceleration
Resource path	GET [coap://[aaaa::b00:f6ff:2d3b:d2c4]:5683/sensors/acceleration

We build and deploy a 6LoWPAN network composed of two 6lowPAN nodes: a wearable device and a 6lowPAN Border Router (6LoBR). 6LoBR plays an important role in the communication inside and outside of our 6LowPAN network. The 6LoBR is responsible for (i) exchange data between the wearable device and cloud services and (ii) provide forwarding and routing capabilities inside the 6LoWPAN network. In this work, the Smart IoT gateway plays the role of 6LoBR.

2.3.4 IoT gateway

In this section, we introduce the functional modules and hardware used to build the IoT gateway.

IoT gateway architecture IoT gateway is the key component of our fall detection system and integrates five main modules: interoperability, data transformation, big data analyzer, emergency alerts handler, and semantic representation of data.

Interoperability IoT gateway acts as a bridge between 6LowPAN network and cloud services, thus enabling the connectivity and seamless communication between all the system's components. It provides protocol conversion functions that include 6LowPAN transition mechanisms IPv6/IPv4 and message translation between CoAP and Message Queuing Telemetry Transport (MQTT) protocols.

Data transformation This performs two functions. On one hand, the module receives the movement data (x, y, and z acceleration values) and performs filtering using a first-order IIR low-pass filter, and, on the other hand, it annotates and maps data in a comma-separated value (CSV) file format.

Each filtered acceleration value is stored locally to be used as input to the big data analyzer module.

Big data analyzer This is responsible for processing and analyzing acceleration values in the x, y, and z axes to detect if these values represent a fall or ADL using ensemble-RF model previously selected. The features of the ensemble-RF model for fall detection building are described below.

Ensemble-RF model It was built leveraging the advantages of feature bagging prediction technique to exhibit a better prediction performance. Bagging method generates many replicates of a predictor and uses them to get an aggregated predictor [64]. Ensemble-RF consists of 10 base models of trees which are independents in terms of decision, each of which tries to predict the problem's objective field. The base models were trained by making bagging replicates (random subset of the features) of the learning set from fivefold cross-validation.

Figure 6 illustrates the results of the classification from one of the 10 base models. This model indicates how important each variable (acceleration measured in the x, y, and z axes) is in fuzzy decision-making capabilities (prediction) of fall types and of the ADLs. Each node represents a classification rule (i.e., IF-THEN rule) to a variable. Branches leading from the node indicate the path(s) that could be taken. Each path has a confidence level associated with it. The thickness of the branches indicates the amount of training data that this path undertook. The leaf node constitutes the decision based on a prediction. Since ensemble uses decision trees, also it can be easily converted to classification rules.

Fall prediction To fall detection, the big data analyzer creates a local instance of the ensemble-RF within the IoT gateway through the REST API provided by BigML. By carrying the processing close to the data source (i.e., the 6lowPAN wearable device used by the older adult), the system reduces the latency and overheads in the network, and as a result, the long lie time is also reduced. Ensemble-RF predicts a fall or ADL by doing plurality vote over the models, i.e., it obtains a class vote from each tree, and then classifies using majority vote, taking as input data the information coming from the transformation module. If the result of prediction is a fall, the system invokes to the emergency alert handler.

Emergency alerts handler sends alert messages of the fall event, along with the geographical position of the elderly person's house, to the groups responsible for the care of the elderly people previously registered in the system using a MQTT broker. Figure 7 shows an example of the notifications sent. The information relating to fall detected also is sent to the cloud services for storage and processing. MQTT is chosen as the candidate protocol to transport the fall data and notifications related because it is a lightweight and secure IoT protocol. MQTT provides end-to-end secure communication and reliability through the SSL (Secure Socket Layer) protocol. It also incorporates several levels of QoS to confirm the delivery of messages, from a non-optimal minimum level (QoS0) to a double-recognition level (QoS2). Since our system is closely related to the elderly person's healthcare, we configure the level QoS = 2 in order to ensure that messages are delivered only once and in a reliable way.

Semantic representation of data The availability, common understanding, and extraction of knowledge of the system data are necessary for the seamless exchange of fall data and related information across different applications and systems. To do so, this module maps data of the falls and ADLs to the semantic data model defined by UniverSAAL platform [65]. UniversAAL is an open-source platform that provides a

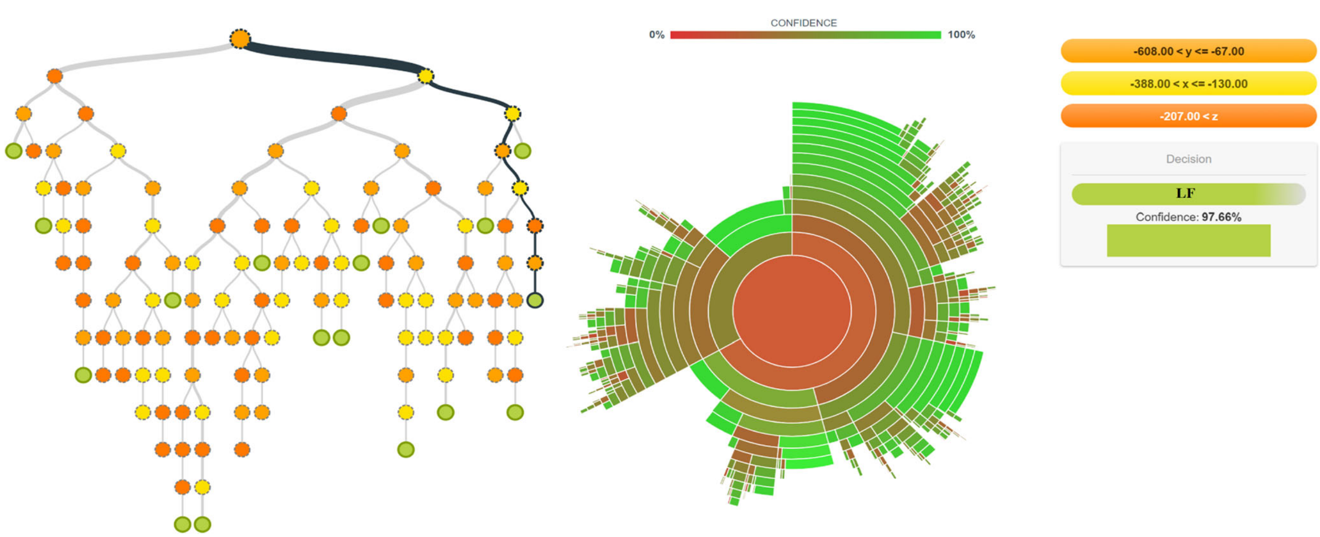

Fig. 6 Classification for one of the decision trees of the ensemble. The left side contains the tree diagram, the center side shows the confidence percentage achieved by decision tree, and the right side shows the decision rule generated by decision tree to recognize a lateral fall

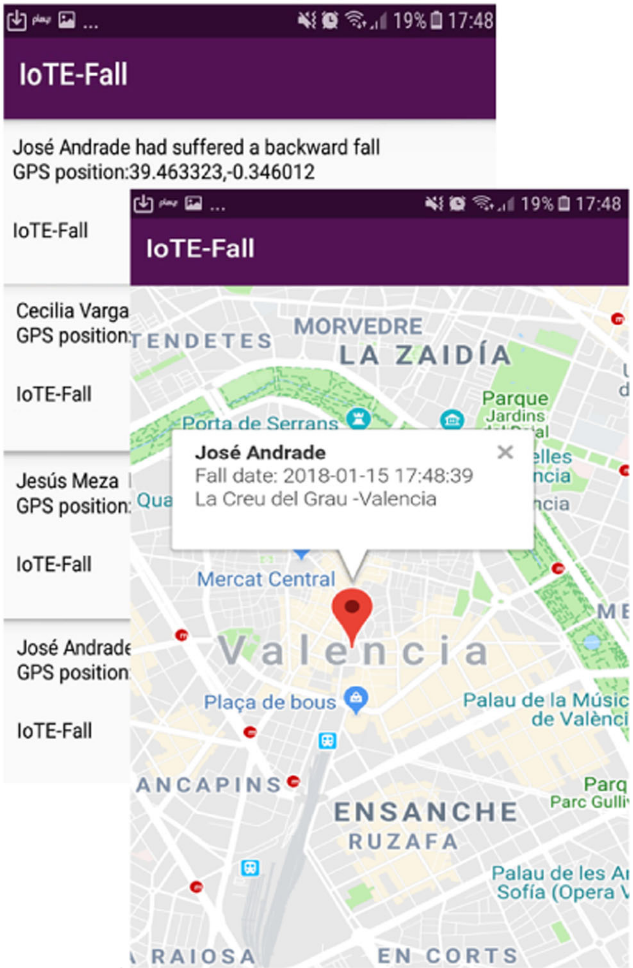

Fig. 7 Example of messages sent to healthcare professionals and family caregiver (back) and location of elderly person's house (front)

standardized approach to facilitate the development of ambient-assisted living (AAL) solutions. The core of UniversAAL is a middleware that contextualizes data and its functionality only through the definition of ontologies. UniversAAL provides a set of ontologies that cover the domains of Smart Environments and AAL and provides semantic services to contextualize the information, i.e., it facilitates the modeling of data, and the representation and serialization of data using Web Ontology Language (OWL) and RDF/Turtle standard, respectively. In addition, UniversAAL enables the exchange of the context information between consumers and providers through the context events; the consumers subscribe to the UniversAAL middleware, who in turn publish the context information to the registered and authorized subscribers.

In this work, we extended the "device" ontology to model the fall and ADL events as shown in Fig. 8. The green boxes represent semantic modules already included in the ontology; meanwhile, the yellow boxes represent the elements extended. Moreover, the UniversAAL middleware was installed in IoT gateway and a publisher module, implemented by an OSGI bundle, was developed. This bundle takes the acceleration data provided by a fall, maps to ontological vocabulary using the extended ontology, and sent it through middleware to interested consumers.

Hardware implementation The IoT gateway is the key component for fall detection designed with a STM32 microcontroller integrated with one expansion board based on the SPSGRF-915, and a Raspberry Pi 3 model B (RPI3) as execution environment, equipped with a 1.2 GHz Quad-Core ARM Cortex processor A53 CPU, 1 GB of RAM, 4 USB ports, and powered by a 3.7 V Lipoly battery. A 16 GB class 10 SD card powered by the Debian Stretch operating system is used to run all of the various functional modules of the system, which are written in Python. Moreover, a low-cost Ublox NEO-6M GPS module is connected to RPI3 for tracking the location of an elderly person's house. The 6LoBR node is defined by the combination of the STM32 microcontroller, and the expansion board based on the SPSGRF-915. The 6LoBR collects data from the wearable device and forward to the RPI3 through a USB serial link. Similar to the wearable device, all operations in the 6LoBR node are executed on "Contiki" OS. RPL is configured in order to allow communication between 6LoWPAN nodes and IoT Gateway. On the other hand, to translate packages coming from 6LoWPAN nodes to IPv6/IPV4 packages and vice versa, a tunneling virtual network adapter is configured in the RPI3 by using *tunslip6* tool running on Contiki OS. Additionally, to retrieve the acceleration values of the wearable device, a CoAP client is implemented in RPI3 using *aiocoap* library based on Python 3 asynchronous I/O library.

2.3.5 Cloud services

The cloud consists of several services, including authentication manager, storage, and ML model. The authentication manager is the entry-point into the cloud; it controls and authorizes the flow of data to and from the system. The storage service preserves the information of the falls and ADLs coming from the IoT Gateway in a meta-model of data in RDF representation format using a semantic database. This storage module was previously implemented by MongoDB, but with the incorporation of UniversAAL framework, we can also make use of any semantic database that supports SPARQL protocol and RDF Query Language that is a SQL-like query language for querying data represented in terms of RDF. Concretely, the Context History Entrepot stores all context events forwarded through middleware to the ontological database in RDF statements.

Finally, once a fall occurs, the model is again created and trained in the cloud using the API REST of BigML via ML model service to be, later on, locally instantiated in the gateway.

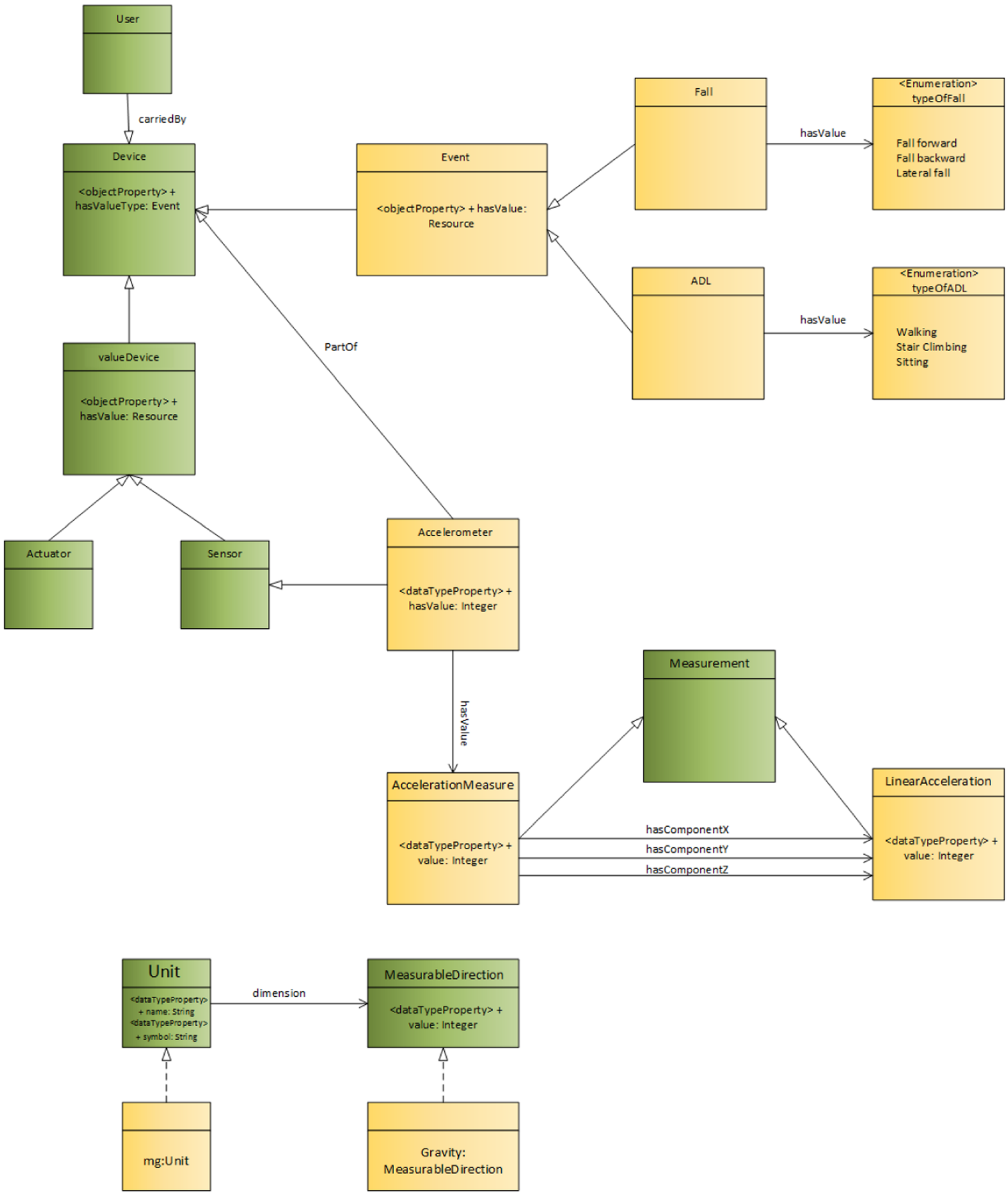

Fig. 8 Ontology extended of UniversAAL for the semantic representation of the events of falls and activities

3 Results and evaluations

3.1 Fall detection results

For performance evaluation of the proposed system, a total of 54 controlled experiments were conducted: 27 experiments of falls and 27 experiments of ADLs. Three subjects (volunteers) between the ages of 40 and 60 years, body mass from 68.7 to 84.6 kg, and height from 1.64 to 1.79 m participated in the experiments. In order to be compliant with GPRD[1] and to ensure the acknowledgment from the participants of the project, prior to these experiments, each subject gave written informed consent to participate in the project and to use the information extracted from the experiments to fulfill its objectives. Each subject performed the falls and ADLs outlined above, and each experiment was repeated three times. Therefore, each subject performed a total of 18 simulations.

[1] https://eur-lex.europa.eu/eli/reg/2016/679/oj

The wearable device was placed on the volunteers' waist because it is considered as the suitable location on the human body to get an optimal performance [66]. The falls were simulated in no particular order. Initially, several tests were performed to tune the system parameters.

The results obtained after the experiments are analyzed using statistical parameters like accuracy, precision, sensitivity, specificity [67]. These parameters are defined by means of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) as follows:

Accuracy corresponds to the correct distinguishing among falls and ADLs and it measures the ratio of correctly predicted samples:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

Precision is the capacity of the system to return only relevant instances (positive predictive value):

$$\text{Precision} = \frac{TP}{TP + FP} \tag{4}$$

Sensitivity (also called recall or gain) represents the capacity of the system to detect actual falls:

$$\text{Sensitive} = \frac{TP}{TP + FN} \times 100 \tag{5}$$

Specificity represents the capacity of the system to only detect falls, which means that events of ADLs such as walking, stairs climbing, or sitting not identified as falls.

$$\text{Specificity} = \frac{TN}{TN + FP} \times 100 \tag{6}$$

- TP is defined as the success of the system to detect a fall or ADL when this one did happen.
- TN is defined as the success of the system to detect the absence of a fall or ADL when this one did not happen.
- FP is the failure of the system detecting a fall or ADL when this one, in reality, did not happen.
- And FN is the failure of the system in detecting a fall of ADL when this one did happen.

Terms true and negative are used to refer to the presence or absence of the condition of interest, fall, or ADL. True positives are the main instances of interest while true negatives are all other instances. Table 4 shows the results achieved by the ensemble model in the fall and ADL detection, which lists the evaluation results of the proposed system for three types of falls and three activities.

Table 4 Consolidated summary of the results achieved by ensemble model in the fall and ADL detection

Target field	Accuracy	Precision	Sensitivity	Specificity
FF	99.80%	98.20%	100.00%	100.00%
BF	99.10%	92.40%	99.40%	99.32%
LF	99.70%	98.10%	99.40%	99.85%
WA	98.90%	98.20%	98.60%	99.50%
SCA	97.10%	93.60%	95.80%	98.34%
SA	97.70%	96.80%	74.40%	99.85%
Average	98.72%	96.22%	94.60%	99.48%

The values of this table are the averages of the results obtained for each type of fall and activity detected. In Table 4, we can see that the recognition accuracy of the system proposed is 98.72%. Since the % ERROR is small (1,28%), according to Han et al. [68], the rules generated by ensemble can be applied to the classification of new data and clearly justifies the decision to train the model with historical knowledge corresponding to the intended audience (i.e., the elderly population). On one hand, since the system is closely bound up to healthcare, the precision is very important to reduce the false alarm, which in our system is 96.22%. On the other hand, the average sensitivity achieved by the recognition of the different types of falls was 99.6%, which indicates that the proposed system recognized almost all of the falls. The system also achieved a high sensibility (99.23%) in the identification of ADLs. This means that there are a relatively small number of cases in which the system considers them as a fall, which reflects that the system is capable of distinguishing falls from all ADLs avoiding the trigger of the alarm system in false positives.

3.2 Comparison with other systems

As stated in Sect. 1, in recent years, several research studies have been conducted to address the fall detection in elderly people. In comparison, with the wearable-based systems that employ machine learning algorithms to fall detection (which are the focus of our approach). Our proposal is the only one that leverages the effectiveness of ensemble algorithm machine learning achieving promising results in fall detection in terms of accuracy, sensitivity, and specificity as shown in Table 5.

Although the algorithm of Tong et al. [32] has an ideal sensitivity (100%) and specificity (100%), the data samples used to test the system does not correspond to the target audience. Thus, the experiment results applied to elders will be different under realistic conditions. In fact, the authors report that the training and testing of the system using large real-world samples of the elders will be conducted in future work. Moreover, this system does not alert in case of a fall, which is

Table 5 Comparison of the performance on work related

Approach	Machine learning algorithm	Accuracy	Sensitivity	Specificity
Our proposal (2018)	Ensemble	98.72%	96.22%	94.60%
Aguiar et al. [16]	Decision tree	92.9%	92.3%	93.2%
Santoyo et al. [19]	SVM	–	93.5%	97.8%
Pierleoni et al. [28]	SVM	95.18%	90.37%	100%
Nguyen et al. [30]	MLP	99.05%	98.26%	99.62%
Özdemir et al. [31]	K-NN	99.91%	100%	99.79%
Tong et al. [32]	HMM	–	100%	100%

a necessary requirement to prevent the long lie and improve the QoL of the elderly.

When we compare with the system proposed by Özdemir et al. [31], we obtain similar outcomes, and the difference between the two algorithms was 3.78 and 5.19% of sensitivity and specificity, respectively. While the accuracy has a negligible difference of 1.19%. However, similar to Tong et al.'s [32] case, the trials used for testing the system correspond to healthy volunteers aged between 21.5 and 27 years. Therefore, the comparison of these performance parameters might be biased toward the former approaches. Moreover, the system shows some pitfalls for real implementation because the authors used six sensor units with three tri-axial devices. The poor usability may pose barriers to adopt this solution. In contrast to the proposal of Özdemir et al. [31], our system uses only the data coming from a 3D-axis accelerometer to fall detection. Thus, our system has fewer data to process and analyzed leading to rapid fall detection, and therefore to reduce the long lie.

Similar to our proposal, the MLP-based system proposed by Nguyen et al. [30] used SisFall dataset for the detection of falls. The results of their fall detector achieved promising results with a specificity of 99.62%. In this case, the accuracy is very close to the approach presented here, with a negligible difference of 0.33% in favor of the fall detector, and it uses trials of both young subjects and elderly to train, test, and evaluate the system. However, the same dataset used in all system stages could cause an overfitting problem so the system will not be able to generalize well enough with new data collected. Moreover, the fall detector does not send notifications when a fall takes place. In contrast, we used only trials of elderly in the training and testing stages and applied fivefold cross-validation to avoid overfitting, while real-time data streams coming from 3D accelerometer sensor is used in the validation stage for detecting falls.

We can note also that our ensemble-based system highlights better performance than the other four related works analyzed [12] [16] [19, 28] in terms of accuracy and sensitivity with a difference in the specificity of 5.4% with Pierleoni approach [28]. However, there are also other important differences between these systems proposed and the system implemented here. Pierleoni et al. [28] employed three tri-axial embedded in a wearable device to fall detection. The data fusion of these sensors may have an impact in the battery life. Moreover, the authors performed two extractions for fall detection: one for identifying the peak and other for detecting the fall orientation, which requires more processing time, and thus may not be supported in real-time applications where rapid processing is desirable to act accordingly when an unusual event is detected.

Santoyo et al. [19] and Aguiar et al. [16] used a smartphone as execution environment for fall data processing. It may lead to shortening the duration of the battery of the smartphone preventing long-time operation of the system. Moreover, the Santoyo approach [19] does not envisage any alarm method to reduce the response time required to assist the elderly. In addition, as comparate to Aguiar's [16] approach, we demonstrated that several decision trees increase the classifier stability and accuracy thanks to plurality voting effectiveness.

In addition, in comparison with all systems analyzed in this study, the semantic interoperability implemented in our system facilitates the data exchange of detected falls in a uniform format with any authorized system.

4 Concluding remarks

The Internet of Things is a new paradigm helping the adult population to improve their quality of life by facilitating a pervasive and more personalized form of care. This study has presented IoTE-Fall system, an IoT system for fall detection of elderly people in indoor environments, based on a Big Data model that uses machine learning processing techniques based on ensemble-RF. We analyzed and tested four machine learning algorithms to detect falls and distinguish them from ADLs. We selected the best suitable algorithm to achieve this goal by comparing the performance, computational requirements, and the area under the receiver operating characteristic (ROC) curve of the classifiers. By exploiting fivefold cross-validation, we have built and tested the classification models from historical knowledge coming from a publicly accessible dataset of falls and ADLs performed by elderly people.

The results reveal that ensemble-RF was the most suitable classifier with average AUC of 0.995, training time of 5.75 s, and testing time of 3.48 s.

The fall detection processing operation using ensemble-RF runs on an IoT gateway that provides edge computing capabilities. To predict the falls, the IoTE-Fall system takes as input the acceleration measured in the x, y, and z axes coming from the movements of elderly volunteers in real time. These values are collected with a 3D-axis accelerometer sensor embedded in a 6LowPAN-based wearable device. The device is placed on the elderly people's waist, and it offers a suitable solution to be used by an elderly person in an indoor environment. The system remotely alerts the healthcare professionals, emergency centers, caregivers and the elderly person's family members in the case that a fall event occurs using QoS mechanisms.

The system' performance was evaluated for recognizing three types of falls: fall forward, fall backward and lateral fall, and three types of ADLs: walking, stairs climbing, and sitting. The average recognition accuracy (98.72%), precision (96.22%), sensitivity (94.60%), and specificity (99.48%) indicate that the proposed system has a high success rate, both in falls and ADL detection.

Likewise, the comparative analysis performed at the end of this work shows the advantages of our fall detection solution over other research projects of the same nature. These advantages involve not only high levels of average accuracy, precision, sensitivity, and specificity, equal or higher to the results brought by other projects but, also, an extra added value with the different services included at the IoT gateway of the architecture operating at the edge of the network. For example, the semantic interoperability service provides a holistic and interoperable IoT solution and enables IoTE-Fall system to coexist and be integrated with other applications or external services. Also, the alarm system service aids effective and in-time decision-making to health professionals by reporting about the fall type and location of the elderly person's house. In addition, thanks to the edge processing of the movement data of the elderly people, our solution also helps to reduce the long lie and consequently improves the quality of life of the elderly people. These services are not contemplated in many of the analyzed studies.

As a summary, this work presents a holistic solution for fall detection whose main advantages include the fast processing and detection at the edge of the network, the enhancement of the detection at each fall with the re-creation of the ML model, and the reliability provided by the IoT protocols involved in the communication. Moreover, the main drawbacks found in the analyzed studies (in Sect. 3.2) that this work attempt to avoid are the use of the same dataset in all the phases of training and evaluation of the ML model, due to that this can cause an overfitting that prevents the system to generalize enough with new data, and the use of data from healthy and/or young volunteers for the system testing, which does not correspond with the target audience, elderly people, in this case. Finally, the use of several tri-axial devices, the wrong usability of these devices can lead to barriers in order to adopt the solution in a real environment, due to the cost of the material and the complexity of these elements.

For future improvements of the system, the integration of more sensors, as heart rate meter, blood pressure meter, among others, and the development of new services related with health is being considered, in order to enlarge system features and benefits.

Funding information Research presented in this article has been partially funded by Horizon 2020 European Project grant INTER-IoT no. 687283, ACTIVAGE project under grant agreement no. 732679, the Escuela Politécnica Nacional, Ecuador, and Secretaría de Educación Superior Ciencia, Tecnología e Innovación (SENESCYT), Ecuador.

References

1. He W, Goodkind D, Kowal P (2016) U.S. Census Bureau, International Population Reports, P95/16-1, An Aging World: 2015. U.S. Government Publishing Office, Washington, DC
2. Bousquet J, Kuh D, Bewick M, Standberg T, Farrell J, Pengelly R, Joel ME, Rodriguez Mañas L, Mercier J, Bringer J, Camuzat T, Bourret R, Bedbrook A, Kowalski ML, Samolinski B, Bonini S, Brayne C, Michel JP, Venne J, Viriot-Durandal P, Alonso J, Avignon A, Ben-Shlomo Y, Bousquet PJ, Combe B, Cooper R, Hardy R, Iaccarino G, Keil T, Kesse-Guyot E, Momas I, Ritchie K, Robine JM, Thijs C, Tischer C, Vellas B, Zaidi A, Alonso F, Andersen Ranberg K, Andreeva V, Ankri J, Arnavielhe S, Arshad H, Augé P, Berr C, Bertone P, Blain H, Blasimme A, Buijs GJ, Caimmi D, Carriazo A, Cesario A, Coletta J, Cosco T, Criton M, Cuisinier F, Demoly P, Fernandez-Nocelo S, Fougère B, Garcia-Aymerich J, Goldberg M, Guldemond N, Gutter Z, Harman D, Hendry A, Heve D, Illario M, Jeande C, Krauss-Etschmann S, Krys O, Kula D, Laune D, Lehmann S, Maier D, Malva J, Matignon P, Melen E, Mercier G, Moda G, Nizinkska A, Nogues M, O'Neill M, Pelissier JY, Poethig D, Porta D, Postma D, Puisieux F, Richards M, Robalo-Cordeiro C, Romano V, Roubille F, Schulz H, Scott A, Senesse P, Slagter S, Smit HA, Somekh D, Stafford M, Suanzes J, Todo-Bom A, Touchon J, Traver-Salcedo V, van Beurden M, Varraso R, Vergara I, Villalba-Mora E, Wilson N, Wouters E, Zins M (2015) Operational definition of active and healthy ageing (AHA): a conceptual framework. J Nutr Health Aging 19(9):955–960
3. Yacchirema DC, Sarabia-Jácome D, Palau CE, Esteve M (2018) A Smart System for sleep monitoring by integrating IoT with big data analytics. IEEE Access, p 1
4. Robie K (2010) Falls in older people: risk factors and strategies for prevention. JAMA 304(17):1958–1959
5. Jrad RBN, Ahmed MD, Sundaram D (2014) Insider Action Design Research a multi-methodological Information Systems research approach. 2014 IEEE Eighth International Conference on Research Challenges in Information Science (RCIS). Marrakech, pp 1–12. https://doi.org/10.1109/RCIS.2014.6861053

6. Chaccour K, Darazi R, El Hassani AH, Andrès E (2017) From fall detection to fall prevention: a generic classification of fall-related systems. IEEE Sensors J 17(3):812–822
7. Min W, Cui H, Rao H, Li Z, Yao L (2018) Detection of human falls on furniture using scene analysis based on deep learning and activity characteristics. IEEE Access 6:9324–9335
8. Ma X, Wang H, Xue B, Zhou M, Ji B, Li Y (2014) Depth-based human fall detection via shape features and improved extreme learning machine. IEEE J Biomed Heal Inform 18(6):1915–1922
9. Yang L, Ren Y, Zhang W (2016) 3D depth image analysis for indoor fall detection of elderly people. Digit Commun Netw 2(1): 24–34
10. Mastorakis G, Makris D (2014) Fall detection system using Kinect's infrared sensor. J Real-Time Image Process 9(4):635–646
11. Kwolek B, Kepski M (2014) Human fall detection on embedded platform using depth maps and wireless accelerometer. Comput Methods Prog Biomed 117(3):489–501
12. Wang Y, Wu K, Ni LM (2017) WiFall: device-free fall detection by wireless networks. IEEE Trans Mob Comput 16(2):581–594
13. Sehairi K, Chouireb F, Meunier J (2018) Elderly fall detection system based on multiple shape features and motion analysis. 2018 International Conference on Intelligent Systems and Computer Vision (ISCV). Fez, pp 1–8. https://doi.org/10.1109/ISACV.2018.8354084
14. Álvarez de la Concepción MÁ, Soria Morillo LM, Álvarez García JA, González-Abril L (2017) Mobile activity recognition and fall detection system for elderly people using Ameva algorithm. Pervasive Mob Comput 34:3–13
15. Fortino G, Gravina R (2015) Fall-MobileGuard: a smart real-time fall detection system. In: Proceedings of the 10th EAI International Conference on Body Area Networks (BodyNets '15). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering). ICST, Brussels, Belgium, pp 44–50. https://doi.org/10.4108/eai.28-9-2015.2261462
16. Aguiar B, Rocha T, Silva J, Sousa I (2014) Accelerometer-based fall detection for smartphones. 2014 IEEE International Symposium on Medical Measurements and Applications (MeMeA). Lisboa, pp 1–6. https://doi.org/10.1109/MeMeA.2014.6860110
17. Kau L, Chen C (2015) A smart phone-based pocket fall accident detection, positioning, and rescue system. IEEE J Biomed Heal Inform 19(1):44–56
18. He J, Bai S, Wang X (2017) An Unobtrusive Fall Detection and Alerting System Based on Kalman Filter and Bayes Network Classifier. Sensors 17:1393. https://doi.org/10.3390/s17061393
19. Santoyo-Ramón JA, Casilari E, Cano-García JM (2018) Analysis of a Smartphone-Based Architecture with Multiple Mobility Sensors for Fall Detection with Supervised Learning. Sensors 18: 1155. https://doi.org/10.3390/s18041155
20. Mao A, Ma X, He Y, Luo J (2017) Highly Portable, Sensor-Based System for Human Fall Monitoring. Sensors 17:2096. https://doi.org/10.3390/s17092096
21. Casilari E, Oviedo-Jiménez MA (2015) Automatic fall detection system based on the combined use of a smartphone and a smartwatch. PLoS One 10(11):e0140929
22. Dias PVGF, Costa EDM, Tcheou MP, Lovisolo L (2016) Fall detection monitoring system with position detection for elderly at indoor environments under supervision. 2016 8th IEEE Latin-American Conference on Communications (LATINCOM). Medellin, pp. 1–6. https://doi.org/10.1109/LATINCOM.2016.7811576
23. Phu PT, Hai NT, Tam NT (2015) A Threshold Algorithm in a Fall Alert System for Elderly People. In: Toi V, Lien Phuong T (eds) 5th International Conference on Biomedical Engineering in Vietnam. IFMBE Proceedings, vol 46. Springer, Cham. https://doi.org/10.1007/978-3-319-11776-8_85. ISBN:978-3-319-11775-1
24. Santiago J, Cotto E, Jaimes LG, Vergara-Laurens, I (2017) Fall detection system for the elderly. 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC). Las Vegas, NV, pp 1–4. https://doi.org/10.1109/CCWC.2017.7868363
25. Malheiros L, Nze GDA, Cardoso LX (2017) Fall detection system and body positioning with heart rate monitoring. IEEE Lat Am Trans 15(6):1021–1026
26. Ethem Alpaydin (2010) Introduction to Machine Learning, 2nd edn. The MIT Press
27. Mezghani N, Ouakrim Y, Islam MR, Yared R, Abdulrazak B (2017) Context aware adaptable approach for fall detection bases on smart textile. 2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI). Orlando, FL, pp 473–476. https://doi.org/10.1109/BHI.2017.7897308
28. Pierleoni P, Belli A, Palma L, Pellegrini M, Pernini L, Valenti S (2015) A high reliability wearable device for elderly fall detection. IEEE Sensors J 15(8):4544–4553
29. Aziz O, Musngi M, Park EJ, Mori G, Robinovitch SN (2017) A comparison of accuracy of fall detection algorithms (threshold-based vs. machine learning) using waist-mounted tri-axial accelerometer signals from a comprehensive set of falls and non-fall trials. Med Biol Eng Comput 55(1):45–55
30. Nguyen LP, Saleh M, Le Bouquin Jeannès R (2018) An Efficient Design of a Machine Learning-Based Elderly Fall Detector. In: Ahmed M, Begum S, Fasquel JB (eds) Internet of Things (IoT) Technologies for HealthCare. HealthyIoT 2017. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 225. Springer, Cham
31. Özdemir TA, Barshan B (2014) Detecting falls with wearable sensors using machine learning techniques. *Sensors* 14(6):10691–10708
32. Tong L, Song Q, Ge Y, Liu M (2013) HMM-based human fall detection and prediction method using tri-axial accelerometer. IEEE Sensors J 13(5):1849–1856
33. SISTEMIC: Research group on Embedded Systems and Computational Intelligence of the Electronics and Telecommunications Department at the Faculty of Engineering, University of Antioquia, "SisFall Dataset." Online. Available: http://sistemic.udea.edu.co/investigacion/proyectos/english-falls/?lang=en. Accessed 2 Feb 2018
34. Rubenstein L (2006) Falls in older people: epidemiology. Risk Factors and Strategies for Prev 35(Suppl 2):ii37–ii41
35. Youn J, Okuma Y, Hwang M, Kim D, Cho JW (2017) Falling direction can predict the mechanism of recurrent falls in advanced Parkinson's disease. Sci Rep 7(1):3921
36. Nevitt S, Cummings MC (2018) Type of fall and risk of hip and wrist fractures: The study of osteoporotic fractures. J Am Geriatr Soc 41(11):1226–1234
37. Karantonis DM, Narayanan MR, Mathie M, Lovell NH, Celler BG (2006) Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. IEEE Trans Inf Technol Biomed 10(1):156–167
38. Khan AM, Lee YK, Kim TS (2008) Accelerometer signal-based human activity recognition using augmented autoregressive model coefficients and artificial neural nets in 2008 30th Annual International. Conf Proc IEEE Eng Med Biol Soc 2008:5172–5175
39. Yoshida T, Mizuno F, Hayasaka T, Tsubota K, Wada S, Yamaguchi T (2005) A wearable computer system for a detection and prevention of elderly users from falling. In: Proceedings of the 12th international conference on biomedical engineering. Singapore, pp 179–182
40. Kangas M, Konttila A, Winblad I, Jamsa T (2007) Determination of simple thresholds for accelerometry based parameters for fall detection. In: 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. Lyon (France), pp

1367–1370. https://doi.org/10.1109/IEMBS.2007.4352552. E-ISSN: 1558-4615
41. Shan S, Yuan T (2010) A wearable pre-impact fall detector using feature selection and Support Vector Machine. In: IEEE 10th International Conference on Signal Processing Proceedings. Beijin (China), pp 1686–1689. https://doi.org/10.1109/ICOSP.2010.5656840. E- ISSN: 2164-523X
42. Lombardi A, Ferri M, Rescio G, Grassi M, Malcovati P (2009) Wearable wireless accelerometer with embedded fall-detection logic for multi-sensor ambient assisted living applications. In: 2009 IEEE Sensors. Christchurch (New Zealand), pp. 1967–1970. https://doi.org/10.1109/ICSENS.2009.5398327. E- ISSN: 1930-0395
43. Aziz O, Klenk J, Schwickert L, Chiari L, Becker C, Park EJ, Mori G, Robinovitch SN (2017) Validation of accuracy of SVM-based fall detection system using real-world fall and non-fall datasets. PLoS One 12(7):e0180318
44. Wang K, Delbaere K, Brodie MAD, Lovell NH, Kark L, Lord SR, Redmond SJ (2017) Differences between gait on stairs and flat surfaces in relation to fall risk and future falls. IEEE J Biomed Heal Inform 21(6):1479–1486
45. Lindholm B, Hagell P, Hansson O, Nilsson MH (2015) Prediction of falls and/or near falls in people with mild Parkinson's disease. PLoS One 10(1):e0117018
46. Fan Y, Levine MD, Wen G, Qiu S (2017) A deep neural network for real-time detection of falling humans in naturally occurring scenes. Neurocomputing 260:43–58
47. Jokanovic B, Amin M, Ahmad F (2016) Radar fall motion detection using deep learning. In: 2016 IEEE Radar Conference (RadarConf). Philadelphia (USA), pp 1–6. https://doi.org/10.1109/RADAR.2016.7485147. E- ISSN: 2375-5318
48. Jankowski S, Szymański Z, Dziomin U, Mazurek P, Wagner J (2015) Deep learning classifier for fall detection based on IR distance sensor data. In: 2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), vol 2. Warsar (Polonia), pp. 723–727. https://doi.org/10.1109/IDAACS.2015.7341398
49. Jokanović B, Amin M (2018) Fall detection using deep learning in range-Doppler radars. IEEE Trans Aerosp Electron Syst 54(1):180–189
50. Shojaei-Hashemi A, Nasiopoulos P, Little JJ, Pourazad MT (2018) Video-based Human Fall Detection in Smart Homes Using Deep Learning. In: 2018 IEEE International Symposium on Circuits and Systems (ISCAS). Florence (Italy), pp 1–5. https://doi.org/10.1109/ISCAS.2018.8351648. E- ISSN: 2379-447X
51. Leu F-Y, Ko C-Y, Lin Y-C, Susanto H, Yu H-C (2017) Chapter 10 - Fall Detection and Motion Classification by Using Decision Tree on Mobile Phone. In: Xhafa F, Leu F-Y, Hung L-LBT-SSN (eds) Intelligent Data-Centric Systems Book. Academic Press, pp 205–237. https://doi.org/10.1016/B978-0-12-809859-2.00013-9
52. Yacchirema D, Suárez de Puga J, Palau C, Esteve M (2018) Fall detection system for elderly people using IoT and Big Data. In: 9th International Conference on Ambient Systems, Networks and Technologies (ANT 2018), Porto (Portugal), available at Procedia Computer Science, vol 130, pp 603–610. https://doi.org/10.1016/j.procs.2018.04.110 E-ISSN:1877-0509
53. Rougier C, Meunier J, St-Arnaud A, Rousseau J (2011) Robust video surveillance for fall detection based on human shape deformation. IEEE Trans Circuits Syst Video Technol 21(5):611–622
54. Stone EE, Skubic M (2015) Fall detection in homes of older adults using the Microsoft Kinect. IEEE J Biomed Heal Inform 19(1):290–301
55. Yuwono M, Moulton BD, Su SW, Celler BG, Nguyen HT (2012) Unsupervised machine-learning method for improving the performance of ambulatory fall-detection systems. Biomed Eng Online 11(1):9
56. Friedman J, Hastie T, Tibshirani R (2001) The elements of statistical learning, vol. 1, no. 10. Springer series in statistics New York, NY, USA. https://doi.org/10.1007/b94608. E-ISBN: 9780387848587
57. Zhang C, Ma Y (2012) Ensemble machine learning: Methods and applications. Springer-Verlag New York, NY. https://doi.org/10.1007/978-1-4419-9326-7. E-ISBN 978-1-4419-9326-7
58. Big ML (2017) Inc. US "Comprehensive Machine Learning Platform". Online. Available: https://bigml.com/features. Accessed 12 Aug 2018
59. Ling CX, Huang J, Zhang H et al (2003) AUC: a statistically consistent and more discriminating measure than accuracy. In: 18th Int'l Joint Conf. Artificial Intelligence (IJCAI), Acapulco (mexico), vol 3, pp 519–524. ISBN:0-7695-2728-0
60. Dai J, Bai X, Yang Z, Shen Z, Xuan D (2010) PerFallD: A pervasive fall detection system using mobile phones. In: 2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops). Mannheim (Germany), pp 292–297. https://doi.org/10.1109/PERCOMW.2010.5470652. E- ISBN: 978-1-4244-6606-1
61. Li Y, Ho KC, Popescu M (2012) A microphone array system for automatic fall detection. IEEE Trans Biomed Eng 59(5):1291–1301
62. Fawcett T (2006) An introduction to ROC analysis. Pattern Recogn Lett 27(8):861–874
63. Pease SG, Trueman R, Davies C, Grosberg J, Yau KH, Kaur N, Conway P, West A (2018) An intelligent real-time cyber-physical toolset for energy and process prediction and optimisation in the future industrial Internet of Things. Futur Gener Comput Syst 79(Part 3):815–829
64. Breiman L (1996) Bagging predictors. Mach Learn 24(2):123–140
65. Hanke S, Mayer C, Hoeftberger O, Boos H, Wichert R, Tazari M-R, Wolf P, Furfari F (2011) universAAL – An Open and Consolidated AAL Platform. In: Wichert R, Eberhardt B (eds) Ambient Assisted Living: 4. AAL-Kongress 2011, Berlin, Germany, January 25–26, 2011. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 127–140. https://doi.org/10.1007/978-3-642-18167-2_10. E-ISBN: 978-3-642-18167-2
66. Gjoreski H, Lustrek M, Gams M (2011) Accelerometer Placement for Posture Recognition and Fall Detection. In: 2011 Seventh International Conference on Intelligent Environments. Nottingham (UK), pp 47–54. doi: https://doi.org/10.1109/IE.2011.11. E- ISBN: 978-0-7695-4452-6
67. Parker C (2011) An Analysis of Performance Measures for Binary Classifiers. In: 2011 IEEE 11th International Conference on Data Mining, Vancouver (Canada), pp 517–526. doi: https://doi.org/10.1109/ICDM.2011.21. E- ISSN: 2374-8486
68. Han J, Kamber M, Pei J (2012) Data Mining Concepts and Techniques, Third Edit. Morgan Kaufmann Publishers in The Morgan Kaufmann Series in Data Management Systems. Waltham (USA). E-ISBN: 9780123814807