IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 30 June 2024, accepted 14 July 2024, date of publication 18 July 2024, date of current version 29 July 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3430395*

RESEARCH ARTICLE

# Streaming Processing for ADL Monitoring in Smart Home Environments

**YUDITH CARDINALE[1], JOSÉ MANUEL NEGRETE RAMÍREZ[2], IRACHE GARAMENDI BRAGADO[1], AND ISMAEL DE FEZ[1]**

[1]Centro de Estudios en Ciencia de Datos e Inteligencia Artificial (ESenCIA), Valencian International University, 46002 Valencia, Spain
[2]ICS-IRIT, Université Paul Sabatier, 31062 Toulouse, France

Corresponding author: Yudith Cardinale (ycardinale@universidadviu.com)

This work was supported by Valencian International University, Spain, through the internal projects in Data Science for Ubiquitous Computing Environments Energy Consumption Aware, under Grant VIU23008 and Grant VIU24004.

**ABSTRACT** Monitoring and detection of Activities of Daily Living (ADL) is a frequent practice to determine the independence of elderly/disabled people in their homes instrumented with a set of sensors, which conform smart home environments. In these scenarios there are two main problems that need to be addressed: the representation of the ADL and the real time processing of the data gathered by the sensor network. Regarding the first aspect, a reliable representation and modeling of ADL to support their automatic representation must consider factors such as human location, presence of physical objects, and time. Concerning the second aspect, the huge volume of data produced by the sensor network at different velocities and with varied formats must be processed as they are generated in order to be able to provide timely responses. In this sense, we extend a previously proposed framework aimed at assessing the level of independence of an elder person living in a smart home by integrating: (i) the HAMSTERS-XL notation to represent ADL and to instantiate independence evaluation models; and (ii) capabilities for batch and streaming processing, based on Big Data engines. We illustrate the suitability and functionality of the extended framework with a use case consisting of a virtual smart home environment and an inhabitant performing five ADL represented with HAMSTERS-XL and the AGGIR grid model. With this experience, we highlight the benefits of task models to represent ADL combined with the use of Big Data tools to process data. We also identify the limitations of the current version of the framework, in terms of the number of people that can be monitored in the same smart home and the deployment in real scenarios, which will be approached in future research.

**INDEX TERMS** IoT sensor networks, activity of daily living (ADL), pervasive health systems and services, AGGIR grid, big data, smart homes.

## I. INTRODUCTION

The population of elderly people is in clear growth. In 2019, it was calculated population of 703 million people aged 65 years or over in a global scale and estimated to elevate to 1.5 billion in 2050 – i.e., one in six people worldwide will be aged 65 years or over [1]. Actually, the number of people suffering from loss of autonomy and depending on others for performing Activities of Daily Living (ADL) is also increasing [2], [3]. This situation raises the requirement for new care delivery structures and mechanisms to improve the quality of care services as well as to decrease medical costs.

However, there are many conditions that prevent easy solutions. Governments of both Europe and the USA are facing a rising number of care dependent elderly people with chronic conditions served in the community, alongside a reduction in the primary care personnel and reductions or changes in public health care finances [4], [5]. Moreover, it is stated that the isolation in a care center is detrimental for the autonomy, the dignity, and the well-being of an individual [6]. To this end, technologies for assisting and surveying the health of residents at home are relevant concerning this

The associate editor coordinating the review of this manuscript and approving it for publication was Vlad Diaconita.

topic and seem to be appropriate solutions; leading to the articulation of assisting and monitoring tools for people at home in a more efficient manner contributing to healthcare solutions that may evolve into ambient and pervasive [7], [8], [9], [10], [11].

These solutions demand reliable representation and modeling of ADL to support their automatic identification from data collected through sensor networks, which can produce a huge volume of data at different velocities and with varied formats. In this regard, the representation and processing of ADL within smart homes can be supported by task analysis including three major aspects [12]: (i) modeling of the tasks; (ii) collection of data; and (iii) data analysis.

Concerning the first aspect, human activity modeling is a difficult aspect; since factors such as people location, detection of physical objects, and time, must be considered [13]. Furthermore, the person-fit environment theory establishes that compatibility between individuals and their environment must be reached in order to cover the requirements of its residents adequately [14]. To this matter, there exist studies focused on modeling basic ADL to measure the independence of elder people living alone at home, such as AGGIR (Autonomie Gérontologique et Groupe Iso Ressources – Autonomy Gerontology Iso-Resources Groups) grid [15] and SMAF (Functional Autonomy Measurement System) [16]. However, there is still a need of proposals regarding ADL representation by means of both task modeling and measurement models in the domain of health and dependence evaluations.

Regarding the second and third aspects, the collection and analysis of data can be supported on Internet of Things (IoT) techniques and Big Data analytic tools, which seem to be appropriate to provide a base platform for frameworks able to recognize, detect, and monitor ADL performed by people within a smart home environment. For real time applications in these scenarios, the information gathered from the sensors must be processed as they are generated in order to be able to provide timely responses (e.g., generate alarms, discover anomalies in the inhabitants' behavior).

To contribute in this context, we extend a previously proposed framework [17], [18] aimed at assessing the level of independence of an elder person living in a smart home. This framework is able to gather sensor data from the instrumented home, detect the ADL performed by the inhabitant based on the sensors' readings, and determine his/her independence level according to the AGGIR grid. However, it presents limitations in two aspects: (i) ADL are defined with a Domain Specific Languages that restricts the representation only to motor tasks; and (ii) it offers only batch processing capabilities to process historical data. To overcome these limitations and extend the functionalities of the framework, we incorporate: (i) a task modeling approach based on HAMSTERS-XL [19] for ADL modeling and to instantiate different independence evaluation models; and (ii) capabilities for batch and streaming processing, based on Apache Spark and Apache Flink engines.

HAMSTERS-XL [19] is an extensible task notation and tool allowing the design, visualization, and simulation of task models; as well as adapting the notation to tasks specific to a device, a context, or a domain. HAMSTERS-XL is an extension of HAMSTERS proposal (Human-centered Assessment and Modeling to Support Task Engineering for Resilient Systems) [20]. The proposed task modeling approach consists of four steps: (i) specification of time and location criteria of each ADL; (ii) identification of type and location of sensors, according to the required events that need to be detected; (iii) orchestration of daily scenario routines to measure the execution of ADL performed by the household resident; and (iv) modeling the ADL based on HAMSTERS-XL.

Concerning to the collection and analysis of data, the extended framework offers batch processing, based on Hadoop File System (HDFS) and Apache Spark, and streaming processing based on Apache Flume, Apache Spark Streaming, and Apache Flink.

We illustrate the suitability and functionality of the extended framework with several use cases in a simulation environment, considering the monitoring of five ADL (i.e., feeding, dressing, toileting, elimination, and transfers), represented with HAMSTERS-XL, and evaluating the level of independence according to the AGGIR model [15]. This independence evaluation model consists of a six-level dependence scale (GIR1 to GIR6), that can be defined based on seventeen three-state variables, which can have one of these values: **A**, for complete independence; **B**, for partial dependence; and **C**, for complete dependence. With this experience, we demonstrate the benefits of representing ADL with task models and the use of Big Data tools for real time analysis. We also highlight the remaining gaps that need to be addressed in this area, as well as the limitations of the current version of the framework, in terms of the number of people that can be monitored in the same smart home and the deployment in real scenarios, which will be approached in future research.

In summary, the main contribution of this research are:

- Review of recent studies focused on ADL monitoring and recognition, and description of recent initiatives that combine ADL monitoring/recognition with Big Data approaches. These recent studies and projects demonstrate the current research interest in this topic.
- A four-step task modeling approach based on HAMSTERS-XL to represent different aspects of ADL (i.e., motor, cognitive, and perceptive actions) and to allow to integrate different independence level evaluation models (e.g., AGGIR, SMAF).
- A framework with capabilities of task modeling and batch and streaming processing, that extends the functionalities of the previous one and provides the base to compare the performance of several processing engines.
- A proof-of-concept to show the suitability and performance of the framework in several use cases.

The rest of the article is organized as follows. Section II provides an overview of the key concepts related to this work concerned to ADL definition and the description of dependence evaluation models. In Section III, studies that focus on the monitoring and recognition of ADL and the use of Big Data technologies in this regard are discussed. Section IV describes task modeling of ADL/IADL by means of HAMSTERS-XL, highlighting its capabilities of representing motor, cognitive, and perception aspects of ADL. The design concepts for extending the previous approach introduced in prior research is outlined in Section V, by providing a broad overview of the architecture of the original framework, as well as the details of the extended proposal. Section VI demonstrates the capabilities of the expanded framework through the application of specific use cases. A discussion regarding the proposed approach is detailed in Section VII, to point out it benefits and limitations. Finally, conclusions are furnished in Section VIII.

## II. PRELIMINARIES

This section describes the main issues addressed in the development of this work.

### A. ACTIVITIES OF DAILY LIVING (ADL)

ADL is a term used in health care to refer to essential and routine activities that most people should be able to perform without assistance.

The inability to perform essential ADL can lead to unsafe conditions and a poor quality of life. Therefore, in many scenarios, such as taking care for the elderly in smart homes, it is necessary to monitor them in order to determine their level of independence. The ADL concept was first proposed by Katz [21], with the purpose of providing a standard for estimating the physical well-being of the elderly and their need for assisted living. The activities considered by Katz that make up the set of ADL are: bathing, dressing, grooming (toileting), transferring, continence, and feeding. In addition, another set of activities, called Instrumental ADL (IADL), was studied in order to assess how well the elderly interact with the physical and social environment, considering the following activities: use of the telephone, shopping, food preparation, housework, laundry, transportation, taking medications, and managing finances [22]. When ADL and IADL can be managed with sensor devices, they are further categorized into low level activities and high level activities [23].

Low-level activities, also known as atomic events, consist of activities that can be detected with the use of a single sensor, such as presence sensor, binary light, radio-frequency identification (RFID) to detect sitting, lying down, standing, moving, rolling activities. Reliable detection of other activities, such as posture and locomotion, has also been shown to be identified with a single sensor, a 3D accelerometer, or a smartphone [24], [25]. Other studies propose the recognition of low-level activities, such as walking, jogging, climbing, descending, sitting, and standing, by means of smartphone sensors [26]; while other works propose the recognition of low-level activities with multiple sensors [27], [28].

High-level activities or complex events are composed of a set of low-level activities, which typically take longer than low-level activities (that is, they can last up to a few hours), e.g., shopping consists of: driving a car, walking, working on a computer, standing in line at a store, and wandering around a store. The importance of this approach lies in how the recognition of low-level activities can be used for the recognition of high-level activities.

Although the recognition of high-level activities is important for the description of an individual's daily routine, research so far has focused mainly on low-level activities from which high level activities can be identified [29], [30], [31], [32], [33], [34].

### B. DEPENDENCE EVALUATION MODELS

By means of data recorded by the sensors, the information needed to detect the behavior of smart home residents can be obtained and, thus, the automatic identification of ADL is possible. Afterward, the degree of dependence of people in the smart home environments can be determined based on a measurement model. In this section, we describe the most relevant models.

#### 1) KATZ INDEX OF ADL

The Katz index of ADL was developed to address the functional status as a measurement of the ability of the elderly to perform ADL independently [21], [35]. The Katz index includes six items describing ADL ordered by difficulty: (*a*) bathing; (*b*) dressing; (*c*) toileting; (*d*) transfer; (*e*) continence; and (*f*) feeding. In order to evaluate the independence of the elderly, a "yes/no" score is given to each item; using 1 point scale of independence and 0 point for dependence. The total score value indicates the level of dependence: (i) 6 indicates full function; (ii) 4 indicates moderate impairment; and (iii) 2 or less implies severe functional impairment.

#### 2) THE LAWTON-BRODY INSTRUMENTAL SCALE

The Lawton Instrumental Scale represents a tool to assess the Instrumental ADL (IADL) such as: (*a*) ability to use telephone; (*b*) shopping; (*c*) food preparation; (*d*) housekeeping; (*e*) laundry; (*f*) mode of transportation; (*g*) responsibility for own medications; (*h*) ability to handle finances [22].

These eight items are measured with 0 for dependence and 1 for independence. Women are evaluated on all the eight areas of functions, whereas for men, food preparation, housekeeping, and laundry are excluded. For women, a score ranges from 0 (low function or dependent) to 8 (high function or independent). For men, the score ranges from 0 through 5 [22], [36]. Moreover, elderly are evaluated according to their highest level of functioning.

**TABLE 1. AGGIR variables [15].**

| Type | Variables |
|---|---|
| Discriminatory | Coherence<br>Orientation<br>Toileting<br>Dressing<br>Alimentation<br>Elimination<br>Transfers<br>Indoor movement<br>Outdoor movement<br>Distant communication |
| Illustrative | Management<br>Cooking<br>Housekeeping<br>Transportation<br>Purchases<br>Medical treatment<br>Leisure activities |

#### 3) THE FUNCTIONAL AUTONOMY MEASUREMENT SYSTEM (SMAF) MODEL

SMAF is a clinical rating scale used in Canada that measures the functional autonomy of elderly patients [16]. It consists of a 29-items rating scale used to evaluate the dependence of the person and access to available social or material resources devoted to disabilities as well the stability of resources. These items are included in five aspects of functional abilities: (*a*) ADL (7 items); (*b*) mobility (6 items); (*c*) communication (3 items); (*d*) mental functions (5 items); and (*e*) IADL (8 items). Such items are evaluated using a function scoring that determines the evaluated dependence: (*a*) *0*: independence; (*b*) $-0.5$: independence but with difficulty; (*c*) $-1$: needs supervision or stimulation; (*d*) $-2$: needs help; and (*e*) $-3$: dependent. Ten of the SMAF items are measured only by using a 4-point rating scale (i.e., *0*, $-1$, $-2$, and $-3$), such as urinary, bowel, and vision.

#### 4) THE AUTONOMY GERONTOLOGY ISO-RESOURCES GROUPS (AGGIR) GRID MODEL

A similar tool to the SMAF model is the AGGIR grid [15]; which is an autonomy assessment tool used in France to measure the independence level of elderly people. AGGIR categorizes autonomy levels to various environmental factors affecting a person's activities and social life. Seventeen activities are considered in the evaluation. Ten of them are considered "discriminatory" variables and apply to the physical environment; they are used to evaluate the level of assistance a person needs to carry on with normal ADL. Seven "illustrative" variables measure the social environment; they are used to evaluate how much assistance a person needs to lead a normal social life. Each variable is categorized by three major states: ***A***: the individual completes alone, spontaneously, habitually, totally, and correctly; ***B***: the individual can complete alone, but not spontaneously, or correctly, or habitually, or partially; ***C***: the individual cannot complete, needs assistance, or must have someone else do the activity. The 10 discriminatory variables and the 7 illustrative variables are enlisted on Table 1.

**TABLE 2. AGGIR Scale dependence levels [37].**

| AGGIR Scale | Dependence Levels |
|---|---|
| GIR 1 | Bedridden or confined to an armchair AND mental faculties severely impaired |
| GIR 2 | Confined OR impaired mental faculties |
| GIR 3 | Help several times a day for ADL's |
| GIR 4 | Loss of autonomy for transferring, sometimes also regarding toileting or dressing, OR mobile but needs help to perform ADLs, including eating |
| GIR 5 | Help for bathing and home care |
| GIR 6 | Autonomous |

The Groupe Iso-Resource (GIR) (Table 2) helps determine if a person is entitled to a benefit as well as determine the level of benefit the person can receive. The GIR score is based on answers to questions or by observation. The calculator assigns a score between 1 (0 percent of autonomy) and 6 (93 percent of autonomy). A score below 4 entitles a person to public assistance: full assistance for a score of 1 and partial assistance for a score of 3. A score of 4 may entitle an individual to some assistance. Scores above 4 do not entitle a person to benefits under the national long-term care program (Allocation Personnalisée d'Autonomie, APA). The scores can be used for other purposes, such as insurance claim evaluation [15].

## III. RELATED WORK

In this section, we first describe some studies dealing with ADL monitoring and recognition, then several recent initiatives that combine ADL monitoring/recognition with Big Data approaches are presented.

### A. ADL MONITORING AND RECOGNITION

With the purpose to asses human activity recognition within a smart home environment, several sensor network based approaches have been proposed aiming attention from atomic events (recognized by a single sensor reading) to high-level or complex events (recognized by a combination of several sensor readings). For a complete solution for human activity recognition, there are several aspects that have to be contemplated, apart from the type of sensors considered and the type of ADL/IADL targeted to recognize. The model or approach used to represent the activities, the method to perform their identification, and the capability of doing it in real time are also aspects that characterize solutions in terms of flexibility and usability. Table 3 shows a summary of a comparative evaluation of recent studies dealing with ADL/IADL monitoring and recognition for elderly people, in terms of **sensor network** (i.e., the types of sensors considered), **ADL/IADL representation** (i.e., how the activities are modeled), the **method of recognition** used to identify the activities, the **type of the recognized activities** (i.e., *Atomic* (A) or *Complex* (C), in real time or not), and the **scope** of applicability.

The **sensor network** considered in most approaches include a variety of ambient sensors (e.g., presence, magnetic, reed, temperature, $CO_2$ sensors) [18], [38], [39], [40], [41], [42], [43] and sensors of mobile devices [44] or wearable devices [26], [27], [28], [30], [45], [46], [47], [48], such as accelerometer, gyroscope, magnetometer. Some other works consider just one type of sensor, such as binary ambient sensors [29], accelerometers [25], Radio Frequent (RF) wireless sensing [49], [50], Infrared (IR) sensors [51], or a Kinect camera to analyze the making-tea process [52]. A combination of different types of sensors are also contemplated in some studies, for example combining ambient sensors and wearable devices [32], or sensor of wearable and mobile devices [31], or wearable devices combined with sensors of a companion robot [53], or even all kind of sensors (ambient, wearable, and mobile sensors) [33]. Our framework, as the work proposed in [33], has no limitation to support any type of sensor; it is able to integrate environment sensors, mobile devices, and wearable sensors, gather data from them and perform batch and streaming processing.

In some of those approaches, the **ADL/IADL representation** relies on modeling sequences of sensor events represented by a set of operators - e.g., the Allen's temporal logical operators [54], such as FOLLOWS and OVERLAPS (also called as SEQ or AFTER) to represent patterns of complex events. With these types of operators it is possible to represent the timespan of activities and their logical order or co-occurrence. Thus, the **methods of recognition** to identify the ADL/IADL are data-driven based on the readings of sensors and this representation of activities with logical operators [41], [47]. Other approaches **represent** the events by means of Domain Specific Languages (DSL) to define the events in terms of logical operators, time restrictions, and relations among the sensor readings, from which the ADL/IADL are recognized (i.e., data driven **recognition method**) [18]; while other works consider task-models-like approaches for **modeling ADL/IADL**, but they also use data driven **recognition processes** [38], [39], [52]. More formal of **ADL/IADL representation** are done with ontologies, which allow both data driven processing [32], [43] and machine learning models [28] to **identify activities**. An approach to **represent activities** based on multiple and incremental fuzzy temporal windows under a fuzzy aggregation is presented in [29]; this provides an accurate representation of recent events from binary data streams and allows the use of Long Short-Term Memories (LSTM) models to **identify ADL/IADL**.

Many approaches do not use **ADL/IADL representation**, but base their **recognition** in machine learning models. Thus, they need a learning process that depends on available datasets of activities. The machine learning models used are varied. Some approaches explore statistical and supervised classification models, such Support Vector Machine (SVN) [45], Linear Discriminant Analysis (LDA) [27], [31], Convolutional Neural Networks (CNN) [25], [42], Decision Trees [28], [50], LSTM [49], Time Delay Neural Network (TDNN) [40], and Dynamic Bayesian Network (DBN) [53]. Many studies do not focus the recognition of activities in just one model, but in a combination of several of them; for example LSTM and CNN [30], [33], Recurrent Neural Networks (RNN) and LSTM [51], K-Nearest Neighbour (KNN) and CNN [48], or even more than two machine learning models [26], [44]. Also, few other works combine supervised and unsupervised models to identify ADL/IADL, as the works presented in [46] and [47], which combine clustering techniques with deep learning models; whilst the approach proposed in [42] combines CNN and data driven processing to refine the identification of ADL/IADL.

In summary, concerning the **representation of ADL/IADL**, few works pay attention to this aspect, mostly based in task-models [29], [38], [39], [52], DSL [18], or ontologies [28], [32], [43]. Because of the advantages of using task models, regarding to flexibility and extensibility for integrating different elderly dependency evaluation models, in this work, we decided to use task models to represent ADL/IADL. However, others models can be used, such as DSL or ontologies, to represent other aspects and knowledge in the domain (e.g., model the distribution of sensors, maintain a knowledge base for further analysis).

Regarding the **method of recognition**, although the current version of our proposed framework is based on a data driven approach, it can be extended by integrating machine learning models in order to extend its functionalities (e.g., recognize more complex ADL/IADL, detect emergency events, implement more advanced independence evaluation models).

Independent of the **sensor network** and the **method of recognition** used to identify activities, most of these approaches are able to recognize *atomic and complex (A/C)* events. Only two works are focused to identify just one activity. The work presented in [25], dedicated to detect fall from single readings of an accelerometer and the study presented in [52], which recognizes the complex activity of making-tea with a Kinect camera and intelligence compliant objects. However, only few of these works are able to monitor and recognize ADL in real time [26], [38], [40], [43], [44], [47], [50], [53]. Based on the variety of sensors supported by our framework, it is able to support the monitoring and recognition of both *atomic and complex* events by processing the data in batch or real time.

Regarding the **scope** of applicability of these works, those studies that claim to monitor ADL/IADL of elderly must necessarily recognize them in real time [26], [38], [40], [43], [44], [47], [50]. Many works are only focused on ADL/IADL identification, not necessarily in real time, and do not propose models to represent them; thus, they focus exclusively in ADL/IADL recognition [28], [30], [31], [33], [42], [46], [48], [49], some others additionally recognize fitness exercises [45], human identification [51], or people isolation [41]; other works focus only on the recognition of specific activities, such as fitness exercises [27], fall detection [25], or making-tea [52]. Our proposed framework is a complete and holistic solution aimed to model, monitor, and recognize ADL/IADL, based on batch and streaming processing; thus, it can be used for many applications in the context of elderly healthcare, in particular when they live alone, or in any other use case that demands identification of activities from sensor readings.

Most of aforementioned works are focused on identifying correctly activities that require monitoring for elderly people. However, there is still a lack of ADL/IADL to be modeled, whereas others consider a subset of specific activities; but most importantly, they are not based on a specific tool, such as the AGGIR grid, to guide the model of ADL in smart home environments to evaluate elderly independence. These limitations can be overcome by introducing ADL task modeling of the elderly inhabitant based on a set of parameters, such as the AGGIR system, the SMAF model, the Katz index, which measure the autonomy of the elderly, as well as allows implementing other welfare and healthcare applications. From this revision, it is evident that more efforts are needed to propose holistic approaches offering the three aspects in ADL/IADL: *modeling/representation*, processing techniques for *monitoring*, and *recognition* in real time, as we propose in this work.

### B. BIG DATA AND ADL RECOGNITION AND MONITORING

Some recent studies have shown the benefits of Big Data technologies in the context of ADL monitoring and recognition in smart home environments. Table 4 summarizes the comparative evaluation of some recent works that propose solutions in this context, in terms of **sensor network**, **ADL/IADL representation**, the **method of recognition** used to identify the activities, the considered **Big Data technologies** for *batch* or *real time* processing, and the **scope** of applicability.

Regarding the **sensor network**, most studies consider smart environments instrumented with IoT devices (i.e., ambient, mobile, and wearable devices) [55], [56], [57], [58], [59]; other works also consider specific appliances, such as TV, washing machine, laptop [60]; and an initiative based only on accelerometers data for monitoring obesity patients [61]. Like the works presented in [55], [56], [57], [58], and [59], our proposed framework support any kind of IoT devices.

It is common that solutions for ADL recognition that consider Big Data technologies do not use formal **ADL/IADL representation**, but rely on learning process from available datasets of activities. In contrast, our proposed solution considers a task-model combined with a DSL to represent ADL/IADL and model other aspects, such as the sensor network and the scenarios to be simulated in the smart environment; moreover, there is the possibility of incorporating other knowledge representation, such as ontologies.

Big Data can improve the existing ADL recognition models by using data that is generated by ambient sensors, mobile devices, or wearable devices which keep track of the ADL performed by the monitored subjects in real time. Thus, the **methods of recognition** are mostly based on specific machine learning models, such as K-means clustering [55], [60], deep learning [61], or admit any machine learning model [58], [59]. Few of these works do not apply machine learning models, but they are mostly data driven approaches, which represent the ADL as timespan of activities [56], [57]. Although our framework bases the recognition process in data driven approaches, its architecture allows integrating machine learning models, which we plan to do in the future.

Several **Big Data tools** are considered in these works. For real time processing, Kafka and Spark are the most popular combination [55], [59], also only Spark Streaming is used [61], while for batch processing, MapReduce or Spark are the most common used engines [55], combined with Cassandra [55], [59], combination of SQL/NoSQL repositories [56], [58], [60], or cloud computing [57] to store the data. The architecture of our proposed framework allows integrating any store media (SQL and NoSQL) and processing engine to allow both batch and streaming processing.

Regarding the **scope of applicability**, besides simple ADL monitoring/recognition, some other benefits that can be obtained from the application of Big Data are early detection and prevention of dementia [55], development of databases that can be used to make clinical decisions in the field of elderly healthcare [55], [56], [61], and for improvement of the life quality of the elderly people [55], [56], [57], [58], [59], [60]. The capabilities of batch and real time processing of our framework allow implementing a variety of applications in this domain and in any other domain based on processing IoT data to monitor or recognize activities or events.

All these studies demonstrate that the integration and the analysis of Big Data datasets are seen as a powerful approach, not only for ADL monitoring and recognition, but for many other services in smart home environments. However, there is still a lack of proposals that combine formal representation of ADL/IADL with Big Data analytics tools, which provides more flexible and intelligent solutions, as the one proposed in this work. In the following sections, we describe our proposed approach.

## IV. MODELING OF ADL/IADL VIA HAMSTERS-XL

Task models consist of abstract descriptions of user activities structured in terms of goals, sub-goals, and actions. Task models enable ensuring the effectiveness of an interactive system – i.e., to guarantee that users can perform their work and can reach their goals. Many instances of task analysis and modeling techniques exist to provide support for the design, development, and evaluation of interactive systems, as well as evaluation of user performance while interacting with the system [19]. In this work, we propose a systematic process to model different aspects of ADL and evaluate the independence level of elderly, based on the task representation capabilities of HAMSTERS-XL.

**TABLE 3.** Works concerning technologies for monitoring and detection of ADL/IADL for elderly people.

| Ref. | Sensor Network | ADL/IADL representation | Method of recognition | Atomic/ Complex | Scope |
|---|---|---|---|---|---|
| Hsu et al. (2018) [45] | Wearable devices | By learning from datasets | SVM | A/C | ADL and exercises recognition |
| Medina-Quero et al. (2018) [29] | Binary ambient sensors | Fuzzy temporal windows | LSTM | A/C | ADL modeling, recognition |
| Parvin et al. (2018) [38] | Ambient sensors | Task models | Data driven | A/C (real time) | ADL modeling, monitoring, and recognition |
| Peng et al. (2018) [46] | Wearable devices | By learning from datasets | K-means clustering | A/C | ADL recognition |
| Sfar & Bouzeghoub (2018) [32] | Ambient, wearable sensors | Dempster-Shafer Theory (Bayesian networks) | Reasoning, data driven, Markov logic network | A/C | ADL modeling, anomaly and ADL recognition |
| Crema et al. (2019) [27] | Wearable devices | By learning from datasets | LDA | A/C | Fitness exercises monitoring |
| Santos et al. (2019) [25] | Accelerometer | By learning from datasets | CNN | Atomic | Fall detection |
| Francillette et al. (2020) [39] | Ambient sensors | Behaviour Tree | Data driven | A/C | ADL modeling |
| Paraschiakos et al. (2020) [28] | Wearable devices | Activities ontology | Random Forest | A/C | ADL modeling, recognition |
| Qi et al. (2020) [44] | Smartphone sensors | By learning from datasets | Several DL models | A/C (real time) | ADL monitoring, recognition |
| Xia et al. (2020) [33] | Ambient, wearable, mobile sensors | By learning from datasets | LSTM CNN | A/C | ADL recognition |
| Chen et al. (2021) [30] | Wearable devices | By learning from datasets | LSTM CNN | A/C | ADL recognition |
| Hartmann et al. (2021) [31] | Wearable and mobile devices | By learning from datasets | LDA | A/C | ADL recognition |
| Negrete et al. (2021) [18] | Ambient sensors | DSL | Data driven | A/C | ADL modeling, recognition |
| Oguntala et al. (2021) [49] | RF wireless sensing | By learning from datasets | LSTM | A/C | ADL recognition |
| Shao et al. (2021) [40] | Ambient sensors | By learning from datasets | TDNN | A/C (real time) | ADL monitoring, recognition |
| Chifu et al. (2022) [47] | Wearable device | Timespan of activities | Clustering, DL models | A/C (real time) | ADL monitoring, recognition |
| Howe et al. (2022) [52] | Kinect sensor | Task model | Data driven | Making-tea | Making-tea monitoring |
| Palimkar et al. (2022) [26] | Wearable sensors | By learning from datasets | Several ML models | A/C (real time) | ADL monitoring, recognition |
| Saeed et al. (2022) [50] | RF wireless sensing | By learning from datasets | Several decision trees | A/C (real time) | ADL monitoring, recognition |
| Yuan et al. (2022) [51] | Ambient IR sensors | By learning from datasets | RNN LSTM | A/C | ADL recognition, human identification |
| Zhao et al. (2022) [48] | Wearable device | By learning from datasets | KNN CNN | A/C | ADL recognition |
| Bouaziz et al. (2023) [41] | Ambient sensors | Timespan of activities | Data driven | A/C | Isolation, ADL recognition |
| Li et al. (2023) [42] | Ambient sensors | By learning from datasets | CNN, data driven | A/C | ADL recognition |
| Giannios et al. (2024) [43] | Ambient sensors | Activities ontology | Data driven | A/C (real time) | ADL modeling, monitoring, and recognition |
| Liang et al. (2024) [53] | Wearable sensors, and companion robot | By learning from datasets | DBN, data drven | A/C (real time) | ADL monitoring, recognition |
| Our framework | Ambient, wearable, mobile sensors | Task model, DSL | Data driven | A/C (real time) | ADL modeling, monitoring, and recognition |

**TABLE 4.** Summary of studies related to ADL recognition based on Big Data technologies.

| Ref. | Sensor Network | ADL/IADL representation | Method of recognition | Big Data technologies | Scope |
|---|---|---|---|---|---|
| Yassine et al. (2017) [60] | Ambient sensors, appliances | By learning from datasets, Bayesian networks | K-means clustering | Batch processing SQL/NoSQL storage | ADL recognition for healthcare apps |
| Moldovan et al. (2018) [55] | Ambient sensors | By learning from datasets | K-means clustering | Spark, Kafka Cassandra (real time and batch processing) | ADL recognition for people with dementia |
| Mokhtari et al. (2019) [56] | Ambient, wearable, mobile sensors | Timespans of activities (behaviour model) | Data driven | Real time and batch processing, SQL/NoSQL storage | ADL monitoring, recognition (multiple smart home apps) |
| Xu and Wu (2020) [57] | Ambient, wearable, sensors | Data driven | Data driven | Batch processing Cloud computing | Welfare elderly apps (smart homes) |
| Zhang (2021) [58] | Ambient sensors | No applicable | Diverse machine learning models | Real time and batch processing | Multiple smart home apps |
| Vajagic and Antic (2022) [59] | Ambient sensors | No applicable | Diverse machine learning models | Spark, Kafka Cassandra (real time and batch processing) | Multiple smart home apps |
| Hurtado et al. (2023) [61] | Accelerometer sensor | No applicable | CNN | Spark Streaming (real time processing) | Obesity patient monitoring |
| Our framework | Ambient, wearable, mobile sensors | Task model, DSL | Data driven | Real time and batch proc, Flume MapReduce, Spark SQL/NoSQL | ADL monitoring, recognition (multiple smart home apps) |

### A. HAMSTERS-XL NOTATION

In the case of ADL in the context of taking care of aged people, users' tasks strongly rely on motor, cognitive, and perceptive abilities and actions, as well as on the abilities for coordinating these actions. This type of ADL are strongly evaluated according to the users' capacity of manipulating physical objects while performing their tasks, as well as their capacity of processing information. Therefore, a task-modeling notation that supports the description of ADL requires embedding elements to represent motor, cognitive, and perceptive actions, as well as elements to represent the temporal ordering of actions and elements to represent manipulated objects and information. The HAMSTERS-XL tool fulfills these requirements by providing notations for representing the tasks that users perform when interacting with systems, and the temporal order of these tasks, as well as notations to represent data, such as devices, information, knowledge, and objects manipulated by users (which can be physical objects or software objects). Thus, HAMSTERS-XL enables to adapt the notation to tasks specific to a device, a context, or a domain [19], [62].

HAMSTERS-XL provides support to represent refined types of user's tasks [63]: motor, perceptive, and cognitive (depicted in Figure 1). Cognitive tasks can also be refined into cognitive analysis tasks or cognitive decision tasks (on the right in Figure 1). These tasks are represented in a hierarchical temporally ordered way, with the intention of supporting the modeling of large sets of user tasks [20], as well as supporting consistency, coherence, and conformity between user tasks and interactive systems [64]. A task model looks like a tree diagram with nodes being either a task or a temporal ordering operator (e.g., "≫" stands for sequence, "[>" stands for deactivation). Thus, the HAMSTERS-XL notation defines a node-based hierarchy where nodes can function as either tasks or temporal operators. At the highest level, the top node signifies the user's primary objective, while lower levels correspond to sub-goals, tasks, and actions, mirroring the structure seen in Hierarchical Task Analysis (HTA) representation [65]. At the highest level of abstraction, the notation allows users to break down goals into sub-goals, further decomposing them into activities. This decomposition process results in the creation of a graphical tree composed of nodes. These nodes serve as the building blocks of the model and can take the form of tasks or temporal operators. Tasks can vary in type and contain various details, including names, information specifics, and criticality levels.

Figure 2 shows an example of such a representation, where arcs between data and tasks represent how the data are used:

- The user main goal (depicted in yellow) identifies the user primary objective.
- The sequence operator ("≫", in orange) reflects a temporal order of likely sub-goals (cyan), tasks, or actions.
- The order-independent operator ("|=|", in purple), means that the tasks occurring at its lower level do not require a sequence to be performed.

- From the right, the arcs between the input device "in D: Input Device 1" (outlined in blue) and the input tasks "Detect start of Task 1" and "Detect end of Task 1" (in red and green, respectively) mean that the "Task 1" requires the input device labeled "Input Device 1".
- The arc between the required duration labeled "Duration: Time duration" (outlined in pink) and the input task "Detect end of Task 1" means that "Task 1" involves a period of time comprehending between 10 min and 20 min when performed.

Moreover, subroutines can be generated. A subroutine is a task that points out to another task model, in order to support the structuring and reuse of models. HAMSTERS-XL tool offers a graphical interface to help users elaborate the tasks representation.

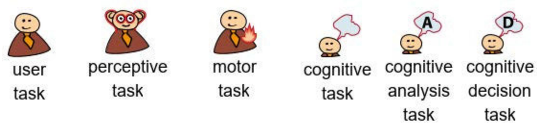


**FIGURE 1.** Types of user tasks in HAMSTERS-XL notation.

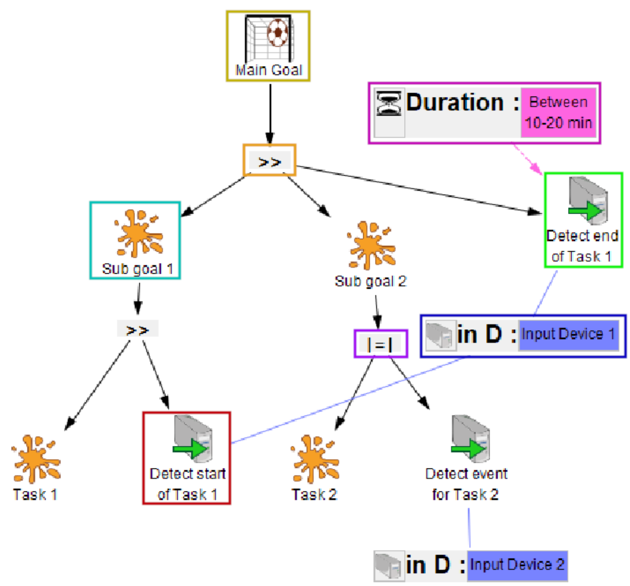


**FIGURE 2.** Graphical representation of ADL modeling based on HAMSTERS-XL notation.

### *B. SYSTEMATIC APPROACH FOR MODELING ADL/IADL WITH HAMSTERS-XL*

In the context of automatic monitoring and recognition of ADL/IADL in smart environments, it is required to represent the sensors, their readings, and the time, location, and sequence of events. Perceptive and cognitive tasks are represented to validate the level of independence, based on criteria, such as the AGGIR model. HAMSTERS-XL notation allows both the ADL representation and the behaviors that describe level of independence.

With the purpose of generating the representation of ADL/IADL for elderly people based on HAMSTERS-XL notations, we propose a systematic process consisting on four steps: (i) specification of the criteria concerning to time and location of each ADL/IADL, which in turn demands the specification of the events that describe each activity; (ii) identification of type, as well as location of sensors, according to the required events that need to be detected; (iii) orchestrate daily scenario routines concerning the performance of ADL by the household resident, which includes physical and virtual interaction with the environment; and (iv) modeling ADL/IADL with the HAMSTERS-XL notations, considering all aspects defined in the previous steps and models of independence, such as AGGIR.

Once the ADL/IADL representation is defined, the model can be integrated in an engine able to process in real time the information gathered from the sensor network of a smart home and recognize the ADL/IADL performed for the inhabitant, based on that representation. In the following section, we describe our framework proposed to do so.

## V. EXTENDED FRAMEWORK WITH HAMSTERS-XL AND BIG DATA TECHNIQUES

This section describes the extension of the framework introduced in previous works [18], [66]. Firstly, the architecture of the original framework is described in general terms; subsequently, the extended proposal considering the HAMSTERS-XL representation and the streaming processing are explained.

### *A. DESCRIPTION OF THE ORIGINAL FRAMEWORK*

The main goal of the framework presented in [18] and [66] is to evaluate the AGGIR metric scale, from data generated by different devices within an intelligent virtual house. The ultimate goal is to identify the independence level of the resident, based on their performance of the ADL/IADL within a smart home environment. The framework is composed of three main modules, as depicted on Figure 3: (i) Descriptor Module; (ii) Simulator Module; and (iii) Analyzer Module. A brief description regarding the features of each module is provided hereafter.

#### 1) DESCRIPTOR MODULE

In order for the user to interact with the framework, the Descriptor Module furnishes a graphical user interface, in which the user sets the parameters for the inhabitant behavior scenarios related to a specific period of time – i.e., it allows defining the parameters for the simulation with respect to the location of sensors, ADL performance location, type of sensors, time, as well as the events needed to be recognized. For the simulation, the user has to specify, through the Descriptor Module, the following aspects: (i) Location map: required to indicate the environment representation by means of the house plan where the sensor network implementation is performed; (ii) Sensor network: focusing on defining information related to the infrastructure of the sensor network environment; for this matter, relevant data must be gathered, such as the inventory of sensors distributed within the smart home, in addition to the location where each sensor is implemented; (iii) Sensor readings format: due to the fact that the data retrieved from sensors are raw format, such data require organization for facilitating their collection and interpretation during the event detection process; (iv) Event conditions and ADL/IADL representation: since conditions are established for triggering the detection of events, such conditions must be defined by the user by providing the ADL/IADL representation; and (v) Event occurrences: once the event conditions have been provided, the event occurrence must be managed in accordance with such aforementioned conditions – i.e., the inhabitant behavior scenarios are defined, which indicate the ADL/IADL he/she will perform in the simulated smart home during the specified period of time.

All these parameters, conditions, and ADL/IADL representations are specified in the Domain Specific Language (DSL) provided by the framework. Non-expert users can specify these configurations through the graphical interface of the Descriptor Module, which automatically generates the DSL representations and store them in XML format. Expert users can provide them directly with the DSL in XML format.

To start the simulation, the Descriptor Module sends to the Simulator Module the sensor network description (i.e., location map and sensor network descriptions) and the inhabitant behavior scenarios (i.e., event occurrences). During the simulation, the defined ADL/IADL (i.e., event occurrences) activate specific sensors in specific moments and spaces, which generate sensor raw data (i.e., sensor readings). Table 5 enlists examples of sensors for ADL detection. The sensor readings and their respective time of the occurrence, represented with the DSL, are stored in a repository also in XML format.

**TABLE 5.** **Smart home sensors.**

| Sensor type | Attributes | Data type | Examples |
|---|---|---|---|
| Electromagnetic sensor | On/Off | Boolean | Cooker/stove; oven;light; switch |
| Proximity sensor | On/Off | Boolean | Sink |
| Capacitive sensor | On/Off | Boolean | Kitchen; counter; chair |
| Magnetic sensor | Open/ Close | Boolean | Refrigerator door; cupboard doors |
| Presence Sensor | On/Off | Boolean | Room occupancy |

### 2) SIMULATOR MODULE

To do not depend on real smart homes and real people, this module offers a simulation environment, in which it is possible to represent a smart home instrumented with a set of sensors, to simulate the ADL carried out by the elderly inhabitant, and to indicate the adequate parameters concerning the daily routine scenarios. To this extent, the iCASA simulation platform [67] was integrated as the framework Simulator Module.

iCASA consists of a smart home simulation tool offering control over several elements of the smart environment (e.g., time, environment, inhabitants, devices, scripting facilities, as well as notification facilities), as presented on Table 6. iCASA was conceived in the medical context by the Adele Research group with the aim of allowing access and control over a digital home environment to developers. Moreover, the iCASA simulator is executed upon the Open Services Gateway Initiative (OSGI) [68] platform for smart-home applications.

**TABLE 6.** **iCASA facilities.**

| | |
|---|---|
| Time | Offers the possibility to slow down, speed up, or stop time during the simulation. Simulation of long-term actions, such as energy consumption, to skip to important actions. |
| Environment | Allows the definition of different zones in a house. Provide an administration interface to modify different physical properties (temperature, luminosity, etc.) of the different zones. |
| Inhabitants | Allows inserting or removing inhabitants from the environment. Inhabitants can move from zone to zone, may be carrying physical devices. |
| Devices | Devices can be simulated or real. At any time, the user can add or remove new simulated devices and modify their localization in the rooms. |
| Scripting facilities | Supports the scripts writing to control the environment. Scripts provide a convenient way to test the applications under reproducible conditions. |
| Notification facilities | iCASA is event-based and is able to notify subscribers of any modifications in the environment. |

### 3) ANALYZER MODULE

The Analyzer Module takes, from the DSL specifications in the XML repository, the information generated by the Simulator Module (i.e., sensor readings after the simulation, consisting of the time when the reading takes place according to the simulator clock, the sensor ID, and the sensed value) and the Descriptor Module (i.e., ADL/IADL representation), in order to recognize and to classify the ADL performed for the inhabitant and evaluate if the AGGIR variables of the case study have been carried out to completion.

To do so, the main components of this module are the Event Detector and the Evaluator. The Event Detector manages the information extracted from the sensor readings (provided by the Simulator Module) to identify events according to the DSL representation of ADL/IADL (provided by the Descriptor Module). The events are identified as atomic and complex actions instances; e.g., if a presence sensor is *on* in room $X$ (atomic), and after a period of time this sensor is *off* (atomic) and another presence sensor is *on* in room $Y$ (atomic); it can be inferred that the action "walk" (complex) was carried out. All atomic and complex events are represented with the DSL. The Evaluator is able to determine whether the AGGIR variables have been performed successfully or not. This module evaluates the achievement of the AGGIR variables, consisting of accomplishing a determined number of events during a specific time lapse – e.g., in order to validate if the dressing AGGIR variable is consummated, conditions such as moving towards the wardrobe at least twice a day must be met.

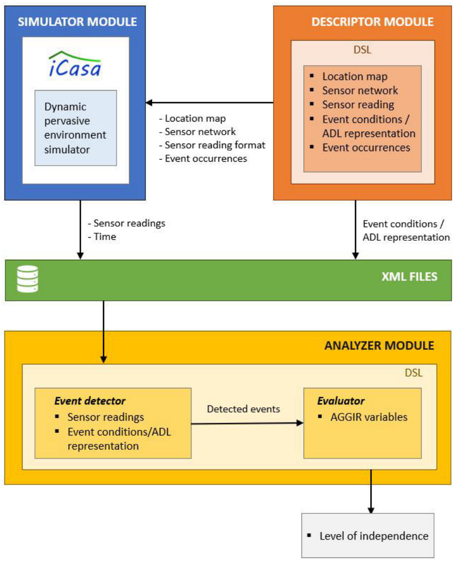


**FIGURE 3.** Previous architecture.

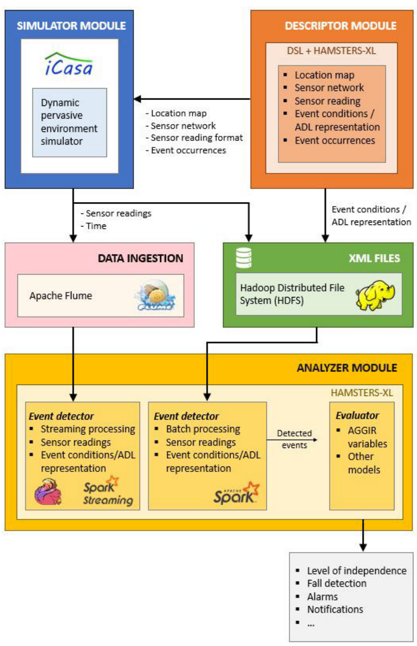


**FIGURE 4.** New architecture.

### B. EXTENDED FRAMEWORK

In order to extend functionalities of the original framework, we improve mainly two aspects: (i) the modeling of ADL/IADL and the integration of several dependence evaluation models; and (ii) data processing capabilities. Regarding the first aspect, we change the DSL representation, which is limited to represent only motor actions, by HAMSTERS-XL task-modeling notation that supports the description of motor, cognitive, and perceptive actions, then offering a more flexible model to represent ADL/IADL. Moreover, with HAMSTERS-XL it is possible to represent other independence evaluation models, different from the AGGIR. For this change, we made modifications in the Descriptor and Analyzer Modules.

The original framework is restricted to evaluate the level of independence of aged people according to the AGGIR model, in batch mode (i.e., from stored historical information). Thus, concerning the second aspect, we improve the batch processing, with HDFS and Spark engine, which provides parallel processing of huge amount of data. Most important, we incorporate the streaming processing capability to be able to process data in real time and extend the applicability of the framework. The streaming processing capability demands three stages: (i) data generation, performed in the Simulator Module based on the iCASA simulator; (ii) data ingestion, a new component based in Apache Flume, integrated to the Analyzer Module; and (iii) data processing, a new component based in Apache Spark Streaming and Apache Flink, also integrated to the Analyzer Module.

Figure 4 shows the new proposal of the framework. In the following, we describe the extension and modification done on each module.

#### 1) DESCRIPTOR MODULE

For the simulation, like in the original Descriptor Module, the user has to specify the location map, the sensor network, the sensor readings format, the event conditions and ADL/IADL representation, and the event occurrences. In this modified version, the event condition, the ADL/IADL representation, and the event occurrences are represented with HAMSTERS-XL; while the rest of the aspects (i.e., location map, sensor network, and sensor readings format) are kept with the DSL representation. Non-expert users can define all these aspects through the graphical interface of the framework and the graphical interface of HAMSTERS-XL, that automatically generate the respective XML files.

#### 2) SIMULATOR MODULE

This module is not impacted by the change of the ADL/IADL representation. We keep the functionality of storing the raw data from sensors in the DSL representation in XML format, for batch processing, as the original version of the framework, but we incorporate Apache HDFS as the storage repository.

For the streaming processing, during the virtual simulation, the inhabitant performs actions inside the house and the sensors generate records that must be analyzed in a subsequent real time process. To this aim, an intermediate component in the Simulator Module has been created in charge of transforming the data issued from the iCASA simulator into a csv format file and send them to the ingestor in the Analyzer Module.

#### 3) ANALYZER MODULE

The data emitted by the Simulator Module is sent to the Analyzer Module for real time and batch processing at the Event Detector. To this extent, we use Apache Flume as the streaming data collection tool and Apache HDFS as a distributed file collection system for the batch processing. Both tools are open source projects of the Apache Foundation.

For the batch processing, as the original version, the Event Detector analyzes the information stored in HDFS related to the sensor readings of a period of time and, according to the event conditions and ADL/IADL representation, identifies the activities performed by the inhabitant in that period of time. These detected ADL/IADL (atomic or complex events) are sent to the Evaluator, which evaluates the level of independence of the inhabitant based in any independence measuring model, such as AGGIR. In contrast to the original version of this module, the extended framework is able to consider any measuring model to determine the level of independence and health of people, thanks to the more flexible and extensible representation provided by HAMSTERS-XL.

For the streaming processing, with the aim of assuring that both the data emission and the data collection are continuously listening, the Apache Flume ingestor is configured through the network. The simulated sensors generate the data that travel through the Internet to Apache Flume, as the intermediate process that generates a dataset with more structured information and transfers it from its place of generation to the data manager, represented by the Event Detector module. This module needs to interpret this information in real time. To do so, the original Event Detector is extended by incorporating streaming processing capabilities, based on Apache Spark and Apache Flink. The objective of providing both streaming processing engines is to allow users/developers to compare/evaluate both engines in their own scenarios and select the one capable of processing in real time a large amount of data generated by the smart home environment and responding in the shortest possible time.

As a general rule, when a large amount of data is generated, it may contain errors or irrelevant data. For this reason, independently of the processing engine, the data are first cleaned and prepared for the analysis process, from which it is possible, not only to detect ADL, but identify anomalies, detect fall, generate alarms/notifications, etc., in real time, according to the proposal of the application. In the following section, we illustrate the functionalities of the extended framework in the context of several use cases.

## VI. USE CASES

In order to evaluate the proposed approach, we present a set of simulated situations, where five of the AGGIR variables are assessed: alimentation, dressing, toileting, elimination, and transfers. The difficulty in determining the values of these variables lays on the fact that some of them can be measured directly, such as opening or closing a door, whereas some others, such as elimination or cook, have a complex context to be determined based on the values of several sensors within a time range. In this section, we first explain how these five ADL are represented with HAMSTERS-XL and then we show their simulation with the extended framework.

### A. ADL MODELING AND REPRESENTATION

To obtain the representation of these ADL, we follow the strategy proposed in Section IV, consisting of the steps described in the following.

#### 1) SPECIFICATION OF TIME AND LOCATION CRITERIA OF ADL

The first step consists of identifying the criteria over the time and location of activities, as well as the identification of the events that have to be detected; in the use cases, all activities related to the five AGGIR variables. The alimentation variable consists on eating prepared foods, and should be performed at least three times per day. The dressing variable assesses the ability to wear clothing, upper and lower body parts of the subject and must occur twice a day. The toileting variable addresses the ability of the resident to groom/wash themselves and should occur at least once every day. The elimination variable monitors the ability to maintain personal hygiene concerning urinary and fecal elimination, which should occur between five to ten times a day. The transfers variables assures the aptitude for getting up, lying down, sitting down in the different household rooms, and should occur more than five times a day [69]. According to the AGGIR model, each ADL is classified as modality **A**, **B**, or **C**: (i) if the inhabitant performs the minimal requirement of the ADL, the activity is classified with modality **A** (ADL complete); (ii) if the ADL is performed less than the minimal requirement, it is classified with modality **B** (ADL partially complete); (iii) if the inhabitant does not perform the ADL, the activity is classified with modality **C** (ADL not complete).

The smart home automation is divided into five zones, each one of these zones represents a room of a realistic household. The design of the house is considered to be the minimum a person needs to live comfortably. Moreover, this resident can carry out their daily routines within the designed zones. For our use cases, the smart home is modeled with the following zones and areas:

- **Bedroom zone**. Area designed to simulate ADL such as sleeping, dressing, or working. The bedroom area has been divided into four sub-areas: (i) **Bedroom**: area comprising the entire area of the room; (ii) **Sleep**: area that includes the bed, intended for rest activities of the inhabitant; (iii) **Chair_bedroom**: area intended for activities related to study, leisure, or work, whose main component is a work table and a chair; and (iv) **Wardrobe**: area intended for activities related to clothing, in which the main component of this zone is a wardrobe.

- **Kitchen zone**. Area designed to simulate ADL such as eating or cooking. The kitchen room has been divided into two areas: (i) **Kitchen**: area that delimits the entire kitchen and includes elements such as the refrigerator and the kitchen; this is an area for carrying out activities related to food preparation; (ii) **Chair_kitchen**: area that includes the kitchen table, destined to carry out activities related to eating.
- **Living room zone**. Area designed to simulate ADL related to leisure and socialization. This area is composed of two areas: (i) **Living room**, which comprises the entire living room area; and (ii) **Armchair**, referring to the sofa/armchair designated area.
- **Hallway zone**. Area designed to detect movement between areas. It is composed of a single area.
- **Bathroom zone**. Zone intended to detect ADL concerning personal hygiene. The bathroom is composed of four areas: (i) **Bathroom**: area comprising the entire bathroom; (ii) **Washbasin**: smaller area whose main component is the sink; (iii) **Toilet**: smaller area whose main component is the toilet basin; and (iv) **Bathtub**: area regarding the bathtub/shower area. ADL related to personal hygiene can be carried out in these areas. The bathroom is an important area in terms of detecting ADL and relating them to a person's level of dependence.

All of the above-mentioned zones intervene in the identification of the transfer ADL.

### 2) IDENTIFICATION OF THE TYPE AND LOCATION OF SENSORS

To detect each event, it is necessary to identify the type and location of sensors that can describe the occurrence of such event. The devices and sensors used for the configuration of the smart home environment are:

- **Binary light**. Provides a 100-watt incandescent light bulb model. The color of the radiation is white monochromatic emission type. The lamp power is fixed at 100 watts and it is considered as a binary sensor, with two states, *on* or *off*.
- **Presence/proximity sensor**. The presence sensor can be used to detect whether a person is in a room or not. It is a binary sensor with two states, either *on* or *off*.
- **Door sensor**. The door opening sensor can be used to detect the opening of different types of doors. It is also a binary sensor, with the *on/off* status.
- **Thermometer**. The thermometer can be used to get the current temperature in a room. The thermometer device returns a Kelvin measurement value for the temperature.
- **Carbon dioxide detector**. This device measures the carbon dioxide ($CO_2$) gas level in the air. It is useful for identifying high concentration of $CO_2$, as well as raising alerts in such as cases.
- **Capacitive sensor**. This sensor can be employed to detect pressure on furniture surfaces, such as chair, armchair, table, bed. With this sensor it is also possible to detect proximity, humidity, or fluid level when using for example the sink, the toilet, the washbasin, or the bathtub.

These types of devices are distributed in the defined zones of the smart home, as follows:

- **Bedroom zone**. The devices indicated in the first row in Table 7 have been included in each area of this zone. These devices are binary, meaning they have two possible states (*on* or *off*). In other words, with the presence sensor defined for the bedroom, it can be established that the person is carrying out an activity related to resting/sleeping as long as the presence sensor state is *on*. On the other hand, if a change is perceived in the wardrobe door sensor, it can be inferred that the subject performs an activity related to clothing.

**TABLE 7.** Devices within the smart home zones.

| Zones | Areas | Devices |
|---|---|---|
| Bedroom | Bedroom | Presence sensor<br>Binary light |
| | Sleep | Capacitive sensor |
| | Chair_bedroom | Capacitive sensor |
| | Wardrobe | Door sensor |
| Kitchen | Kitchen | Presence sensor<br>Thermometer<br>Carbon dioxide detector<br>Binary light |
| | Chair_kitchen | Capacitive sensor |
| Living room | Living room | Presence sensor<br>Binary light |
| | Armchair | Capacitive sensor |
| Hallway | Hallway | Presence sensor<br>Binary light |
| Bathroom | Bathroom | Presence sensor<br>Binary light |
| | Washbasin | Capacitive sensor |
| | Toilet | Capacitive sensor |
| | Bathtub | Capacitive sensor |

- **Kitchen zone**. The sensors located in this zone are detailed in the second row in Table 7. With the purpose of detecting whether the resident has entered or left the area in which they are located; the binary light state will be either *on* or *off* depending on the presence of the person in the kitchen.
  The thermometer device is employed to monitor the temperature variations. These changes will be logged and can be analyzed in order to include temperature regulation systems concerning heating control or alarms. $CO_2$ sensors measure the $CO_2$ concentration in the room. As with the thermometer, these records can be analyzed for including a process helping to warn if a dangerous situation is present for the subject, or even for regulating the entry of gas, or opening doors activating motors to provide ventilation to the area. Also, from

these non-binary devices, information about the person's control in ADL can be extracted, such as cooking, as well as their level of dependence; e.g., if a person leaves the gas open while cooking, it will be advisable to provide help in this regard.

- **Living room zone**. The sensors in the living room zone are shown in the third row in Table 7.
  This room can be considered as the simplest in terms of the design of areas and sensors. However, the activities performed in this room are related to relevant ADL regarding the detection of dependence level. In order to introduce an operating logic for the heating system, temperature sensors can be included in this area, as exemplified in the kitchen area. Moreover, capacitive sensors can be included to detect possible falls.
- **Hallway zone**. The sensors in this zone are detailed in the fourth row in Table 7.
  Once the analysis of the information received by the sensors in this zone is performed, ADL related to movement can be established: transfers made between zones.
- **Bathroom zone**. The sensors in this zone are enlisted in the fifth row in Table 7. With these sensors it is possible to detect if the inhabitants are performing activities related to their hygiene.

The goal of defining more detailed areas with digital devices enables the identification of ADL more specifically, since the activity will be detected within an area that provides better accuracy.

### 3) ORCHESTRATE THE SCENARIOS

The scenarios for our use cases are related to an elderly indoor daily routines regarding the accomplishment of the AGGIR variables. For this purpose, each simulation consists of an inhabitant of the smart home interacting within the different defined house areas, whose ADL indicate if he/she is an independent person (i.e., the resident performs basic ADL correctly) or needs assistance (i.e., the resident does not perform basic ADL). Hence, his/her simulated ADL within the smart home activate the different sensors, whose readings are analyzed to identify such ADL and, according the AGGIR measurement model, to determine his/her level of independence.

### 4) MODELING THE ADL WITH HAMSTERS-XL

HAMSTERS-XL model provides task models consisting of abstract descriptions of user activities structured in terms of goals, sub-goals, and actions [19]. With the aim to assess our proposal, we show the representation of the five AGGIR variables considered in our use cases (i.e., elimination, alimentation, dressing, toileting, and transfers):

- **Modeling elimination**. In order to perform the `Elimination` ADL, the evaluated person must achieve the following sequence of tasks (depicted in Figure 5): *interactive input task*, conformed by a sequence of a *user motor task* and an *input task* (tasks described under the *sequence operator* "≫"):
  1) `Approach toilet basin` (*interactive input task*), refined by a sequence ("≫") of a *user motor task* and an *input task*:
     - `Approach toilet basin` (*user motor task*).
     - `Detect entering into toilet basin area` (*input task*) relying on the `device toilet basin presence sensor`.
  2) `Detect presence in toilet basin area` with a `duration` from 3 to 15 minutes (*input task*) and relying on the `device toilet basin presence sensor`.

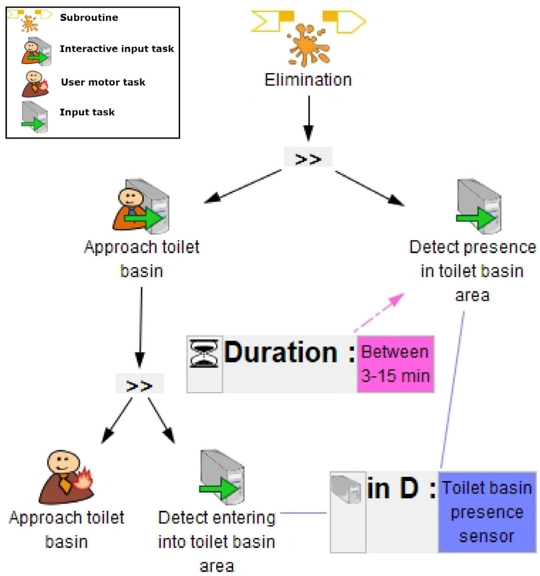


**FIGURE 5. Description of the user tasks to "elimination" with HAMSTERS-XL notation.**

- **Modeling alimentation**. Figure 6 depicts the task model with the stages for the alimentation ADL. The `Alimentation` goal is a sequence ("≫") of two tasks:
  1) `Enter kitchen area` (*interactive input task*) refined by the sequence ("≫") of a *user motor task* and an *input task*):
     - `Enter kitchen area` (*user motor task*).
     - `Detect entrance in kitchen` (*input task*), by means of the `device kitchen presence sensor`.
  2) The next tasks occur under the *order-independent operator* "|=|":
     - `Detect presence in kitchen` (*input task*), by means of the `device kitchen presence sensor`, in a `duration` between 10-20 min.
     - The *concurrent operator* ("|||") indicates the performance of the following tasks:

- `Sit on chair to take meal`, which is an *interactive input task*, described by a sequence ("≫") of: `sit on chair` (*user motor task*) and `detect pressure on chair` (*input task*), relying on the `device chair pressure sensor`.
- `Detect chair pressure sensor is on` (*input task*), relying on the `device chair pressure sensor`, in a `duration` between 10-20 min.

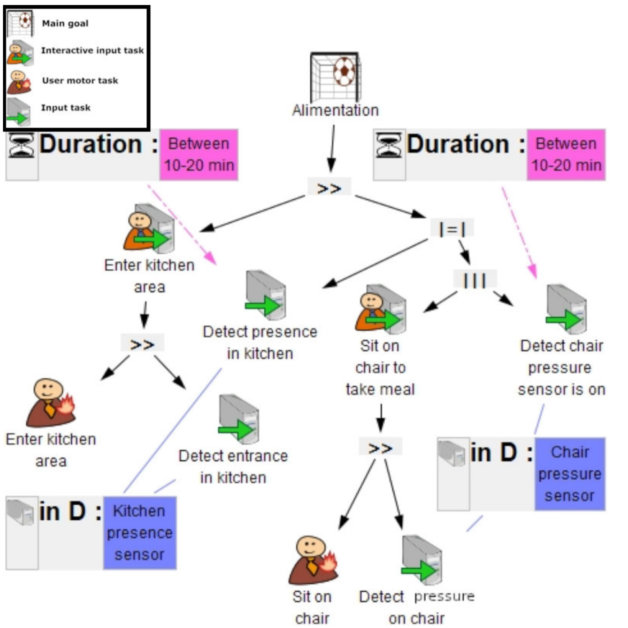


**FIGURE 6.** Description of the user tasks to "alimentation" with HAMSTERS-XL notation.

- **Modeling dressing**. The dressing ADL follows a structure similar to the alimentation ADL, as shown in Figure 7: the goal is a sequence ("≫") of two tasks, in which the first one is an *interactive input task* and the second one is a series of tasks that occur under the *order-independent operator* (|=|). In this case, the *interactive input task* is refined by a sequence ("≫") of a *user motor task* (`Approach wardrobe area`) and an *input task* (`Detect entering into wardrobe area`, by means of the `device wardrobe presence sensor`). The next series of tasks are combinations of *interactive input tasks* and *input tasks* detected by means of the corresponding devices and with duration restrictions, as shown in the right branch of the `Dressing` goal depicted in Figure 7.
- **Modeling of the toileting ADL task**. The requirements of `Toileting` are similar to `Alimentation` and `Dressing` in terms of the structure of the tasks: the goal is a sequence ("≫") of two tasks, in which the first one is an *interactive input task* and the second one is a series of tasks that occur under the *order-independent operator* (|=|). Figure 8 shows its description in terms of

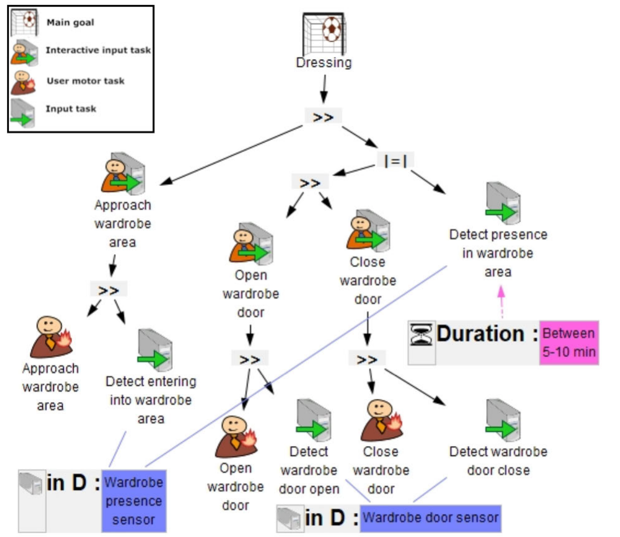


**FIGURE 7.** Description of the user tasks to "dressing" with HAMSTERS-XL notation.

HAMSTERS-XL notation. The first task of the sequence ("≫") is an *interactive input task* refined by a sequence ("≫") of a *user motor task* and an *input task*; while the tasks under the *order-independent operator* (|=|) are combinations of *interactive input tasks*, *user motor tasks*, and *input tasks* supported in several devices and with time restrictions (see the right branch in Figure 8).

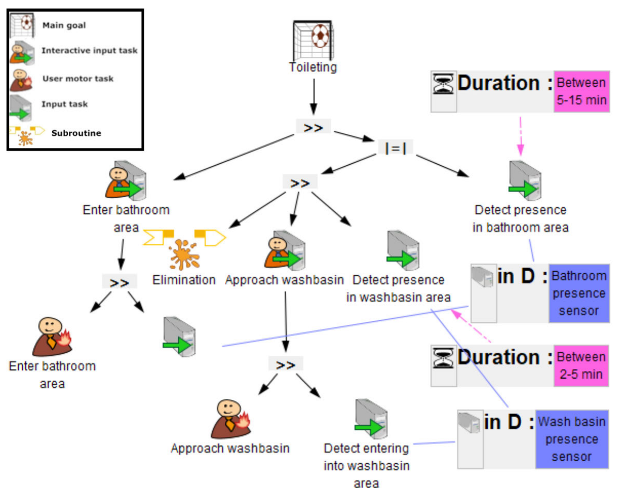


**FIGURE 8.** Description of the user tasks to "toileting" with HAMSTERS-XL notation.

- **Modeling transfers**. In contrast with previous ADL representation, the `Transfer` goal is represented with tasks under the *order-independent operator* "|=|", that can be performed in not matter what order or frequency, as depicted in Figure 9; such tasks are conformed by sequences regarding an *interactive input task*, as well as an *input task* each (both tasks described under the *sequence operator* "≫"):

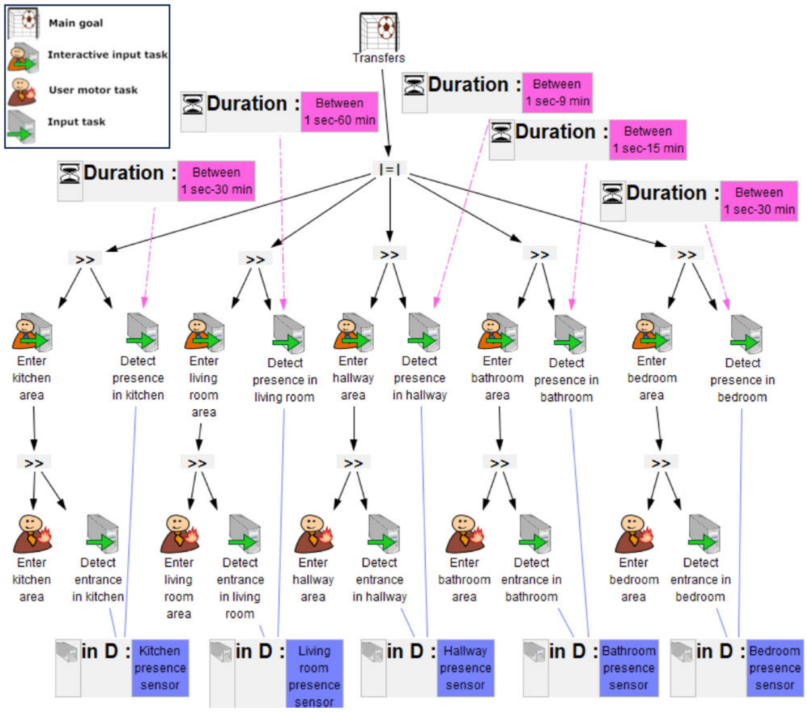


**FIGURE 9.** Description of the user tasks for "transfers" with HAMSTERS-XL notation.

- Detecting presence in `* area`[1]:
  - `Enter * area` (*interactive input task*) conformed by a sequence ("≫") of a *user motor task* and an *input task*:
    - `Enter * area` (*user motor task*).
    - `Detect entrance in * area` (*input task*) relying on the `device * area presence sensor`.
  - `Detect presence in * area` (*input task*) relying on the `device * area presence sensor`, in a `duration` between 1 s to 60 min (according to the task).

### B. SIMULATION WITH THE FRAMEWORK

Once the ADL are represented and the scenarios are specified, the simulation can be performed. In the following, we describe the process on each module of the framework.

#### 1) DESCRIPTOR MODULE

All specifications and representations of ADL and scenarios described in the previous section are elaborated through the Descriptor Module based on HAMSTERS-XL and the DSL. As a starting point for the ADL simulation, a script is also configured in order to be read by the iCASA simulator. The script is an XML file based on the DSL notations, consisting of the following blocks:

- **Areas**. These blocks contain information that characterizes the different areas; e.g., area name, location in the house, and default values for various features, such as temperature. As previously mentioned, there are thirteen designed areas: bedroom, sleep, chair_bedroom, wardrobe, kitchen, chair_kitchen, living room, armchair, hallway, bathroom, washbasin, toilet, and bathtub.
- **Devices**. In this block, the different devices, their id, and their location are configured. As mentioned before, the types of devices used are: binary lights, presence detectors, door opening sensors, thermometers, gas detectors ($CO_2$, CO), capacitive detectors.
- **Inhabitants**. In this block, the residents who live virtually in the smart home are configured. By default, they are located in an existing area. For the purpose of our work, we consider a single person on each simulation.
- **Activities**. In this block, an increment of time between ADL is parameterized, as well as the value for the devices at that time, based on HAMSTERS-XL notations. This block represents the routines of the subject who might live in the actual smart home environment. For the use cases, we have simulated two scenarios for two kind a behaviors: (i) **Grandfather 1**: an individual that performs at least the minimal requirement for each ADL (i.e., his/her ADL are classified as **A**); and (ii) **Grandfather 2**: a person who partially carries out the ADL (i.e., ADL classified as **B** or even **C**).

These scripts are sent to the Simulator Module to start the simulation.

[1] `* Area` represents `kitchen area, living room area, hallway area, bathroom area, bedroom area.`

### 2) SIMULATOR MODULE

As previously stated, the Simulator Module of the framework is based on the iCASA simulator. From the script received from the Descriptor Module the simulator creates the smart home environment with different areas, to which different devices and sensors are added. Figure 10 shows the design of the virtual smart home environment for the use cases.

The inhabitant starts to perform activities around the smart home as indicated in the script (i.e., simulated movements of a resident during a given time). While the ADL are performed by the virtual inhabitant, the data from the sensors are gathered to be sent to the Analyzer Module for processing in streaming or to be stored in HDFS. In both cases, the final data are built with the information collected from each device, using the following format:

*Date, device id, property, property value, area*

This format facilitates data analysis by the processing engines. With the objective of calculating the time at which an event occurs in a device, the process starts from the initial time defined in the simulation, then it adds the time increments that appear defined in the delay. Next, the device data are read and extracted: id, property, and property value. The device id is extracted and the area where it appears defined is searched in order to include such information in the dataset. To this extent, the result is a file with the .csv extension.

Thus, data represented with this format are sent to the data processing managers at the Analyzer Module, through Apache Flume for real time processing and stored as .csv files in HDFS, for further batch processing. Aiming to simulate the daily routine of the inhabitant, the next ADL are considered:

- Wake up in the morning.
- Move to the bathroom for toileting.
- Move to the bedroom to get dressed.
- Move to the kitchen for breakfast.
- Move to the bathroom for elimination.
- Move to the living room to rest.
- Move to the kitchen to cook and eat.
- Move to the bathroom.
- Move to the living room to rest.
- Period away from home.
- Move back home.
- Move to the bathroom.
- Move to the kitchen for dinner.
- Move to the living room for resting.
- Move to the bathroom.
- Move to the bedroom to sleep.

### 3) ANALYZER MODULE

As we stated before, this module has the ability to process the data gathered from sensors both in batch mode and in real time; the Event Detector receives these data and perform the analysis accordingly.

*Event Detector: Pre-processing and ADL identification:*

For both batch and streaming processing, the received data are processed in a pipeline consisting of:


**FIGURE 10.** **The smart home environment design by means of iCASA simulator.**

- *Initialization and creation of the data structure*: according to the processing engine, the execution environment is initiated and the data structures are created.
- *Data filtering*: several transformation are applied to the data for cleaning and preparation.
- *Data grouping*: several transformations are applied to group the data by device (sensor), value *on*, and by area; then, the data are sorted by occurrence time.
  In this use case, the data are grouped by area and by *on* status, which represents that a binary device has been switched from *off* to *on*.
- *ADL recognition*: according to the ADL representation using HAMSTERS-XL notation and the values of the sensors, the ADL performed by the inhabitant are identified. These detected events are sent to the Evaluator.

For the batch processing, based in Apache Spark, the data are taken from HDFS. The *initialization* consists of the creation of the Spark context and the Resilient Distributed Datasets (RDD). *Data filtering* is done with a series of Spark transformations over the RDD to perform the pre-processing phase (data cleaning and preparation); then, other series of Spark transformations are applied for *grouping data* by device, *on* value, and area. From these RDD, the ADL performed by the inhabitant are detected. The whole pipeline is illustrated in Figure 11.

For the streaming processing, the data are received from Apache Flume. For the analysis, we incorporate two processing engines: Apache Spark Streaming and Apache Flink, that can perform in real time the data cleaning and preparation, as well as the ADL detection to be sent to the Evaluator for determining the level of independence of the inhabitant. Although in this use case, we only show ADL recognition in real time, with this capability it is possible to implement other functionalities, such as anomalies detection, falls detection, alarms, and notifications programming.

Both pipelines are similar to the one of the batch processing. With Spark Streaming, the *initialization* consists

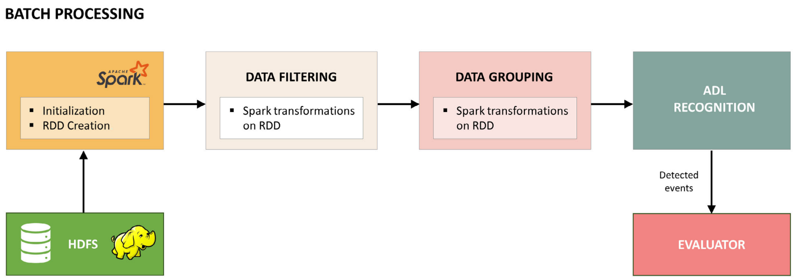


**FIGURE 11.** Batch processing with Spark and RDD.

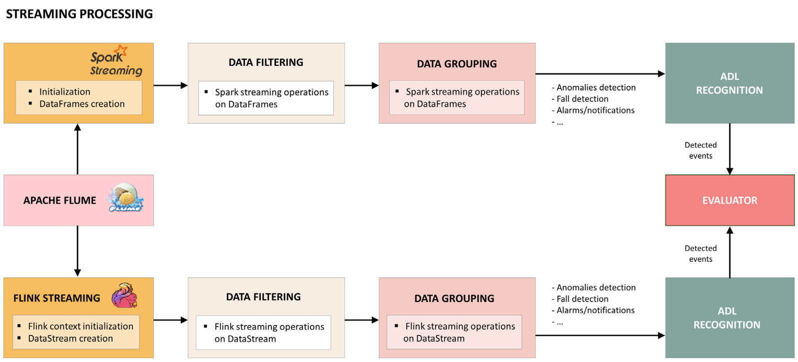


**FIGURE 12.** Streaming processing with Spark Streaming over DataFrames and with Flink over DataStreams.

of creating the spark streaming context and the Data Frames. The *streaming context* allows the use of checkpoints, which are recording points, periodically saving the data (defined by the user); thus providing fault tolerance. A Data Frame is the abstraction provided by Spark with the aim of representing a stream of data. With Flink, the *initialization* also consists of creating the execution environment, in which the streaming program is executed. It runs on a Java virtual machine (JVM). The environment provides methods to control the execution of the job (such as setting parallelism or fault tolerance/checkpointing parameters). DataStream structures are created from the sensors readings received from Flume. *Data filtering* and *data grouping* are performed with operations over the respective data structures according to the respective streaming processing engine. Figure 12 shows both pipelines for Spark Streaming and Flink.

For the use cases, five ADL related to the AGGIR grid scale are recognized: alimentation, dressing, toileting, elimination, and transfers. Thus, the *ADL recognition* process is executed on the respective engine, as follows:

- Alimentation: To identify the alimentation ADL, the data generated by the capacitive sensor installed in the kitchen area is used; more specifically, in the lower level area of the kitchen, called chair_kitchen. The objective is to find a pattern in the dataset as follows: (i) search for the time, set as $t_1$, when the capacitive sensor in the chair_kitchen area changes from *off* to *on*; (ii) look for a time $t_2$, greater than $t_1$, in which the capacitive sensor of the chair_kitchen zone changes to the *off* state; (iii) then, it is interpreted that the individual has been using the kitchen to feed himself. Figure 13(a) shows the logic for the detection of the alimentation ADL, which is intended only for purposes of illustrating the functionality of the Analyzer Module. Moreover, the alimentation ADL can be modeled more precisely, e.g., with sensors in the refrigerator door, on the stove, and on the dining table/chair.

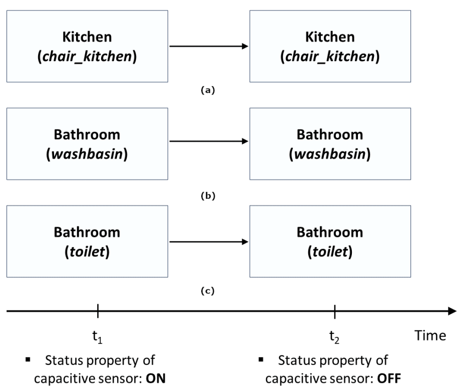


**FIGURE 13.** ADL detection logic: (a) Alimentation; (b) Dressing; (c) toileting.

- Toileting: Based on the data generated by the bathroom sensors, a pattern is sought in order to identify that the inhabitant performs the toileting ADL. For this purpose,

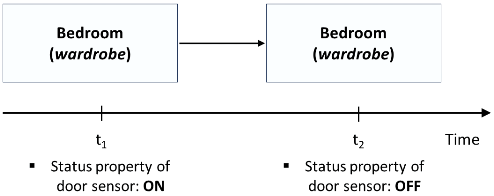


**FIGURE 14.** ADL detection logic: dressing.

the data is filtered by the so-called washbasin area, as shown in Figure 13(b). The steps explaining the procedure are as follows: (i) a time is required, set as $t_1$, in which the capacitive sensor must be activated; (ii) a time $t_2$ greater than $t_1$ is sought in which the capacitive sensor switches to the *off* state; (iii) then, it is considered that the person has carried out the toileting ADL. This ADL can be better modeled by considering a sink faucet activation sensor.

- Elimination: From the data generated by the bathroom sensors, a pattern will be sought with which to identify that the person performs the activity of going to the bathroom or elimination. To this extent, the data is filtered by the area of the toilet basin, the so-called area toilet: (i) a time is sought in which the capacitive sensor is activated, set as $t_1$; (ii) a time $t_2$ greater than $t_1$ is sought in which the capacitive sensor switches to the *off* state; (iii) the person is then considered to have performed the elimination ADL (Figure 13(c)). A more complete modeling of this ADL can consider a toilet flush activation sensor.
- Dressing: From the data generated by the sensors in the bedroom, a pattern must be found enabling to identify that the inhabitant performs the activity of getting dressed. For this matter, the data is filtered by the so-called wardrobe area. The process is as follows: (i) a time is sought in which the door sensor is activated, set as $t_1$; (ii) a time $t_2$ greater than $t_1$ is sought in which the door sensor switches to the *off* state; (iii) then it is considered that the person has carried out the ADL regarding getting dressed (Figure 14). As for the alimentation ADL, this ADL can be modeled with more sensors.
- Transfers: From the data generated by the sensors in the areas of the smart home environment, a pattern will be sought with which to identify that the person performs the transfers ADL. The procedure for achieving the transfers recognition are as follows: (i) a searching for an active presence sensor must be executed; (ii) at that moment, its recovered time is set as $t_1$; (iii) the area in which the presence sensor is located is set as $area_1$; (iv) subsequently, the presence sensor for $area_1$ will be in a deactivated state at a time $t_2$ if the person has moved to another area; (v) at that instant, a new search for an active presence sensor in a different area is triggered; (vi) if time $t_2$ is a later time than $t_1$, and $area_2$ is different from $area_1$, it can be concluded that the person has moved from $area_1$ to $area_2$ (Table 8 and Figure 15).

**TABLE 8.** Transfers logics.

| Area 1 ($area_1$) | Area 2 ($area_2$) | Conditions | |
|---|---|---|---|
| Presence sensor value in $area_1$, status *on* | Presence sensor value in $area_2$, status *on* | Time 1 ($t_1$)>Time 2 ($t_2$) | |
| Time 1 ($t_1$) | Time 2 ($t_2$) | Presence sensor value in $area_1$ at time $t_1$: status *on* | Presence sensor value in $area_1$ at time $t_2$: status *off* |
| | | Presence sensor value in $area_2$ at time $t_1$: status *off* | Presence sensor value in $area_2$ at time $t_2$: status *on* |

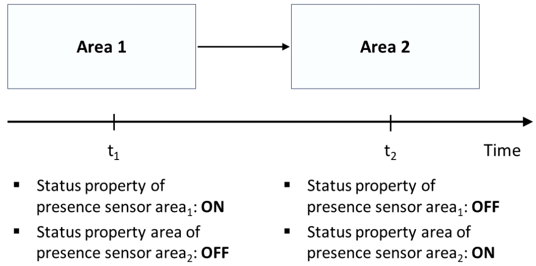


**FIGURE 15.** ADL detection logic: transfers.

According to the established routine, it can be detected that the subject has transferred among the following areas:

- From the bedroom to the bathroom.
- From the bathroom to the bedroom.
- From the bedroom to the kitchen.
- From the kitchen to the bathroom.
- From the bathroom to the living room.
- From the living room to the kitchen.
- From the kitchen to the living room.
- From the living room to the bathroom.
- From the bathroom to the living room.
- From the living room to the kitchen.
- From the kitchen to the living room.

*Evaluator: Determine Level of Independence:*

The ADL identified by the Event Detector are received by the Evaluator to assess the level of independence of the inhabitant of the smart home, according to the AGGIR grid model. To do so, the AGGIR calculator [70] is applied to get the AGGIR score and determine the level of independence. Table 9 shows the results obtained in both simulations. For **Grandfather 1**, we simulated 210 days of activities, in which he performed the five ADL as the minimal requirements, thus obtaining a modality **A** or **B** (in average) for each one. Thus, he reached more than 93% of score, which positions him in group GIR 6 as a totally **Autonomous** person. Meanwhile, for **Grandfather 2**, we also simulated a period of 210 days during which he performed partially the ADL, obtaining, in average modality **B** and **C** for the five ADL, as shown in Table 9. Thus, he was classified in group GIR 4 (**Loss of autonomy**), with a score of 43%. These results are obtained the same independently of the processing engine (Spark, Spark Streaming, and Flink) and demonstrate the functionality of the system: for both scenarios, the system is able to correctly determine the level of dependence of the corresponding inhabitant.

**TABLE 9. Summary of the results of simulations.**

| Simulation aspects | Grandfather 1 | Grandfather 2 |
|---|---|---|
| Alimentation score | A | B |
| Dressing score | B | C |
| Transfers score | B | C |
| Elimination score | A | B |
| Toileting score | A | C |
| AGGIR score | 93% (GIR 6) | 43% (GIR 4) |
| Level of dependence | Autonomous | Loss of autonomy |

In order to compare the performance of the real time processing engines, we measured the total execution time of processing the whole simulation in several scenarios. Although the current functionality of the framework to determine the level of dependence of the inhabitant only demands batch processing (i.e., the person must be monitored for a period of time, in which it is produced historical data to be analyzed further), we generated data from simulations for 21, 210, and 2100 days, with 967 records per day, to be processed in real time. Each record is a sensor reading in the format *Date, device id, property, property value, area*. All tests were executed in a pseudo distributed cluster of 4 nodes installed in an Intel® Core™ i3 processor 14100T, 4.4 GHz, 4 cores, with Ubuntu 18.04.

Table 10 shows the total execution time to process the respective number of records in real time, as well as the throughput in terms of number of records per 200 ms, as this is a good measure to consider real time response [71], [72]. Spark Streaming can process, in average 116 records every 200 ms, whilst Flink is able to analyze 274 records every 200 ms. An event or ADL can be detected with few records: from 2 records for very simple ADL (e.g., presence in a room) to 20 records for more complex events (e.g., emergency event). Thus, these results show the capability of detecting events in less than 200 ms for both processing engines. Actually, in a more powerful cluster and using more efficient data structures, such as Data Streaming instead of RDD for Spark Streaming, the performance can be improved.

Having several processing engines in the framework allows developers to compare their performance for scenarios under the same conditions and help them in the decision of select the most appropriate according to the specific requirements.

## VII. DISCUSSION

In the digital age, a large amount of data is generated in ubiquitous computing environments, as smart homes, populated by sensor networks, as well as mobile and wearable devices. Nowadays, it is becoming easier and more feasible to create this type of environment, since the price of these devices is affordable and their installation is simple. In those cases, these devices generate a large amount of data that should travel over the Internet through an extraction process to then be processed. Thus, it is possible to obtain structured and connected information with which to be able to serve people, by adding an adequate treatment of the data, as we do in this work in the context of ADL recognition for monitoring elderly in smart homes.

Actually, there exist commercial tools that offer a set of sensors, along with an application able to recognize a few types of ADL and generate alerts, such as Canary Care,[2] Essence,[3] LyoTech,[4] and Wimonitor.[5] Besides being private and close solutions, these tools consider few and fixed type of sensors, leading to the recognition of few and limited ADL, they are not capable to process high volume of data, they do not use formal models to represent ADL, many of them do not scale to multiple smart homes, and they cannot be integrated into more general and complete solutions, as the solution proposed in this work.

In the context of ADL recognition for healthcare, it is particularly important to pay attention to the representation of ADL and the methods of recognition, as well as the processing techniques. However, most studies developed in this context consider only partial aspects; they focus on representation models, such as DSL, ontologies, task models, but neglect the recognition methods and the processing of huge amount of data either in batch or real time modes; other works focus their efforts on developing recognition approaches, mostly based on machine learning models, but neglect the representation of ADL leaving it up to the learning process of such models, that in turn depends on available datasets and do not consider any Big Data tool for batch or streaming processing; whilst other studies concentrate their efforts on proposing frameworks capable of collecting, storing, and processing huge amount of data with Big Data tools, but neglect the ADL model representation.

In this work, we demonstrate the feasibility and suitability of combining all these aspects. Moreover, this experience gives the opportunity of extracting the current limitations and some lessons learned.

From the perspective of ADL modeling, the use of formal representations provides many advantages that open a variety of opportunities to implement smart applications in the context of healthcare for elderly people. In particular, task models, as HAMSTERS-XL, allow representing cognitive and perceptive conditions of ADL, besides motor tasks, which provide more flexibility to integrate, for example, different dependence evaluation models, such as AGGIR and SMAF. DSL, ontologies, and timespan representations of ADL are limited to motor tasks. Therefore, our framework provides such as facility to integrate different dependence evaluation models to assess the level of independence of

[2]https://www.canarycare.co.uk/
[3]https://www.essence-grp.com/
[4]https://www.polimi.it/en/scientific-research/research-at-the-politecnico/technology-transfer/spin-off/lyotech
[5]https://www.wimonitor.it/wimonitor/it/

**TABLE 10.** Performance comparison between Spark Streaming and Flink.

| **Simulation time** (days) | **Data size** (records) | **Spark Streaming** Total | Throughput (records by 200 ms) | **Flink** Total | Throughput (records by 200 ms) |
|---|---|---|---|---|---|
| 21 | 20.307 | 37 sec | 110 | 15 sec | 270 |
| 210 | 203.070 | 344 sec | 118 | 146 sec | 278 |
| 2100 | 2.030.700 | 3348 sec | 121 | 1470 sec | 276 |

elderly people (e.g., Katz Index, Lawton-Brody Instrumental Scale, SMAF model), apart from AGGIR, since their criteria can be represented by HAMSTERS-XL. Moreover, several independence evaluation models can be combined to create new ones, as suggested in the study described in [70], that presents the association of values regarding the ADL performed within the AGGIR grid model and other dependency evaluation models. But the advantages are not limited to this. The possibility of representing perceptive, cognitive, and motor conditions for interaction of people with the environment and with other systems, that HAMSTERS-XL provides, brings forth many other applications, such as prevention and early detection of diseases, detection of dementia or Alzheimer, detection of wrong behaviors. With HAMSTERS-XL it is possible to model and represent different behaviors of elderly people to compare, evaluate, and recognize deviations of expected normal behaviour. Moreover, other ADL representation techniques can be integrated into the framework, such as DSL or ontologies. Actually, in the current version of the framework, we keep a DSL to model the scenarios and to describe the sensor network.

From the perspective of Big Data technologies, nowadays it is well known that they offer numerous advantages and opportunities in the field of healthcare in smart home environments, normally combined with other technologies, such as Internet of (Health) Things, Medical Cyber–Physical Systems, and Machine Learning. Big Data concepts have changed the way raw data are gathered, shared, utilized, and transformed into information, which is the base for the design, improvement, assess, and development of a big variety of application in the domain [73]. Batch processing frameworks normally provide easy ways for capturing, collecting, and storing data for further analysis. In this context, applications such as ADL monitoring and recognition, remote patients monitoring, early detection of diseases or wrong behaviors are possible to develop. However, for other kind of applications, the streaming processing is demanded. Data must be analyzed as they are produced, in order to generate responses or actions in real time (e.g., detect emergency events, detect falls, generate alarms/notifications). Our framework offers both batch and streaming processing and provides the base for evaluating and comparing several processing engines under the same conditions.

From the perspective of ADL recognition methods, even though some studies base this task on data driven methods, it seems that the trend is to use machine learning models [74], [75], [76], [77]. Big Data and Artificial Intelligence are being considered the pinnacle of technological advancements. Many recent works demonstrate that intelligent healthcare systems have adopted Big Data and Artificial Intelligence approaches to develop a huge variety of healthcare applications [78]. Big Data benefits and challenges related to the application of the classical machine learning algorithms on several study cases (including ADL recognition), have been presented in many works [79], [80], [81]. Nonetheless stream processing, online learning, and deep learning represent promising research areas that might ensure a better coverage for the analysis of ADL/IADL.

Although the current version of our framework does not provide machine learning models for ADL recognition or other applications, it can be used to generate datasets, automatically labeled, that can be used to train, validate, and test machine learning models. Different scenarios, showing normal or abnormal behaviors, can be modeled at the Descriptor Module using DSL and HAMSTERS-XL that are simulated into iCASA simulator, from which the sensor readings are taken and the repositories are built. Since there exist other simulation tools intended to simulate ADL and generate datasets, such as PerSim 3D [82], IE Sim [83], OpenSHS [84], SESIM [85], and others [39], [86], as a future work, we plan to evaluate the possibilities to integrate them into our framework, in order to improve the Simulator Module. With more flexible and complete simulators, our framework will be able to support more type of sensors, simulate a wider variety of scenarios considering multiple inhabitants, and build more extensive datasets useful to evaluate more machine learning models. Moreover, we also plan to integrate machine learning models to extend the functionalities of the framework to other applications that need to process, both in batch and real time, raw data from smart homes in the domain of smart healthcare systems.

The proposed framework offers a simulation tool that allows developers to design and test smart home based applications without the need of a real instrumented house and people. As a simulation tool, it offers graphical interfaces at the Descriptor Module for non expert users to design the simulation scenarios and the ADL notations. These graphical interfaces automatically generate the respective XML files to be used by the Simulator and Analyzer modules. However experienced users can define the simulation scenarios using directly the DSL and HAMSTERS-XL notation. Additionally, as a simulation tool, the framework offers a platform to asses and compare different processing engines, both batch and streaming, to support the selection of the most appropriate before its actual deployment in real scenarios. Thus, for a real implementation, only the Analyzer Module has to be deployed in a cluster with the respective batch or streaming processing engines, which can be used to manage several smart homes.

The aim of our current solution is focused on monitoring an elder person living alone in a smart home in order to determine his/her level of independence; thus, the current version is limited to detect the ADL performed by one inhabitant. However, it is useful to be able to detect the presence of more than one inhabitant; for example, to detect visitors or to extend the aim of the framework to consider multiple permanent inhabitants. An easy solution is to consider wearable identification devices (e.g., watch, bracelet, band), that are currently supported by the framework, to track ADL of each individual. As a future work, we will also approach this issue.

## VIII. CONCLUSION

In this work, we approach the monitoring and detection of Activities of Daily Living (ADL) in smart home environments considering two aspects: the representation of ADL and real time processing of data generated by sensor networks. To do so, we have extended a previous proposed framework that assesses the level of independence of an elder person living in a smart home environment by integrating a more powerful and flexible model to represent ADL and integrating Big Data analytic tools.

Concerning ADL representation, we propose a methodological process to model ADL with HAMSTERS-XL, an extensible task notation tool, that allows modeling cognitive, perceptive, and motor activities, as well as diverse elements, such as human location, physical objects, and time. We have extended the framework to support task modeling based on HAMSTERS-XL. This integration allows the representation of ADL and the instantiation of several independence evaluation models, which in turn extends the possibilities of implementing a wide range of applications for welfare and healthcare of elderly people in smart environments.

Regarding data processing, Big Data analytics tools are proposed as a suitable solution for processing, in both batch and real time, large volume of data generated by sensor networks. These tools provide a base platform for recognizing and monitoring ADL in smart homes, which constitutes the starting point for developing many applications in the domain of healthcare for elderly people. Batch processing is enriched in the extended framework by integrating HDFS and the Spark engine to reach parallel and distributed data processing, enabling the analysis of historical data. Streaming processing is implemented with Apache Flume for data ingestion and Apache Spark Streaming and Apache Flink for stream data processing. Both engines are suitable for real time processing of data generated by the sensor network and providing timely responses for services that require so. Furthermore, the availability of both streaming processing engines in the extended framework enable users and developers to assess and compare the capabilities of each tool in their specific scenarios. This allows selecting the most suitable engine that can handle real time processing of the substantial volume of data generated by smart home environments and provide quick responses. The objective is to provide flexibility and empower users to make informed decisions based on their requirements, ensuring efficient and timely processing of data for effective monitoring and management of smart home systems.

The functionalities and suitability of the extended framework are demonstrated through use cases in a virtual smart home, where an inhabitant performs five ADL tasks, including feeding, dressing, toileting, elimination, and transfers. Task models are modeled using HAMSTERS-XL based on the AGGIR grid model.

We are currently working on extending our framework in several aspects, such as: (i) the Simulator Module, by integrating other simulators that complement the functionalities of iCASA, for example to be able to manage more than one inhabitant in the smart environment and to generate a wider variety of ADL datasets; and (ii) the Analyzer Module, by integrating machine learning models to have more sophisticated and precise analysis of the generated data. Actually, with the current version of the framework we can generate datasets to train, validate, and test such as intelligent models, but this functionality can be extended with the integration of other simulators.

We also plan to gather data from a real home environment equipped with smart devices and sensors. This would enhance the range of information collected by the sensors and enable real time data processing to be performed. By utilizing real time data, the study could provide more comprehensive insights and potentially extend more practical implications. Also, we will consider the combination of ADL tasks modeling with machine learning techniques to implement more applications in the domain of elderly healthcare.

## REFERENCES

[1] United Nations, Department of Economic and Social Affairs. (2020). *World Population Ageing*. United Nations Publication. [Online]. Available: https://www.un.org/development/desa/pd/sites/www.un.org.development.-desa.pd/files/undesa_pd-2020_world_population_ageing_highlights.pdf

[2] N. C. Tay, T. Connie, T. S. Ong, A. B. J. Teoh, and P. S. Teh, "A review of abnormal behavior detection in activities of daily living," *IEEE Access*, vol. 11, pp. 5069–5088, 2023.

[3] M. Karim, S. Khalid, A. Aleryani, J. Khan, I. Ullah, and Z. Ali, "Human action recognition systems: A review of the trends and state-of-the-art," *IEEE Access*, vol. 12, pp. 36372–36390, 2024.

[4] T. Bienkowska-Gibbs, S. King, C. Saunders, and M.-L. Henham, *New Organisational Models of Primary Care to Meet the Future Needs of the NHS: A Brief Overview of Recent Reports*. Santa Monica, CA, USA: RAND Europe, 2015.
[5] D. S. Kringos, W. G. Boerma, A. Hutchinson, and R. B. Saltman. (2015). *Building Primary Care in a Changing Europe*. World Health Organization. Regional Office for Europe, European Observatory on Health Systems and Policies. [Online]. Available: https://apps.who.int/iris/bitstream/handle/10665/154350/9789289050319-eng.pdf?sequence=3
[6] B. Bouchard, *Smart Technologies in Healthcare*. Boca Raton, FL, USA: CRC Press, 2017.
[7] C. Ramos, J. C. Augusto, and D. Shapiro, "Ambient intelligence—The next step for artificial intelligence," *IEEE Intell. Syst.*, vol. 23, no. 2, pp. 15–18, Mar. 2008.
[8] S. Blackman, C. Matlo, C. Bobrovitskiy, A. Waldoch, M. L. Fang, P. Jackson, A. Mihailidis, L. Nygard, A. Astell, and A. Sixsmith, "Ambient assisted living technologies for aging well: A scoping review," *J. Intell. Syst.*, vol. 25, no. 1, pp. 55–69, Jan. 2016.
[9] V. Plantevin, A. Bouzouane, B. Bouchard, and S. Gaboury, "Towards a more reliable and scalable architecture for smart home environments," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 7, pp. 2645–2656, Jul. 2019.
[10] N. Dua, S. N. Singh, S. K. Challa, V. B. Semwal, and M. S. Kumar, "A survey on human activity recognition using deep learning techniques and wearable sensor data," in *Proc. Int. Conf. Mach. Learn., Image Process., Netw. Secur. Data Sci.* Cham, Switzerland: Springer, 2023, pp. 52–71.
[11] L. D'arco, G. Mccalmont, H. Wang, and H. Zheng, "Application of smart insoles for recognition of activities of daily living: A systematic review," *ACM Trans. Comput. Healthcare*, vol. 5, no. 1, pp. 1–34, Jan. 2024.
[12] P. Johnson, *Human Computer Interaction: Psychology, Task Analysis and Software Engineering*. New York, NY, USA: McGraw-Hill, 1992.
[13] Q. Ni, I. Pau de la Cruz, and A. B. García Hernando, "A foundational ontology-based model for human activity representation in smart homes," *J. Ambient Intell. Smart Environments*, vol. 8, no. 1, pp. 47–61, Jan. 2016.
[14] A. E. M. van Vianen, "Person-environment fit: A review of its basic tenets," *Annu. Rev. Organizational Psychol. Organizational Behav.*, vol. 5, no. 1, pp. 75–101, Jan. 2018.
[15] E. Dupourqué, S. Schoonveld, and J. B. Bushey, "AGGIR, the work of grids," *Long-Term Care News*, vol. 32, pp. 1–11, Jan. 2012.
[16] R. Hebert, R. Carrier, and A. Bilodeau, "The functional autonomy measurement system (SMAF): Description and validation of an instrument for the measurement of handicaps," *Age Ageing*, vol. 17, no. 5, pp. 293–302, 1988.
[17] J. M. N. Ramírez, P. Roose, M. Dalmau, and Y. Cardinale, "An event detection framework for the representation of the AGGIR variables," in *Proc. Int. Conf. Wireless Mobile Comput., Netw. Commun.*, 2018, pp. 153–160.
[18] J. M. N. Ramírez, P. Roose, M. Dalmau, Y. Cardinale, and E. Silva, "A DSL-based approach for detecting activities of daily living by means of the AGGIR variables," *Sensors*, vol. 21, no. 16, p. 5674, Aug. 2021.
[19] C. Martinie, P. Palanque, E. Bouzekri, A. Cockburn, A. Canny, and E. Barboni, "Analysing and demonstrating tool-supported customizable task notations," *Proc. ACM Hum.-Comput. Interact.*, vol. 3, pp. 1–26, Jun. 2019.
[20] C. Martinie, P. Palanque, and M. Winckler, "Structuring and composition mechanisms to address scalability issues in task models," in *Proc. IFIP Conf. Hum.-Comput. Interact.* Berlin, Germany: Springer-Verlag, 2011, pp. 589–609.
[21] S. Katz, "Studies of illness in the aged. The index of ADL: A standardized measure of biologic and psychologic function," *JaMa*, vol. 185, pp. 94–99, 1963.
[22] M. P. Lawton and E. M. Brody, "Assessment of older people: Self-maintaining and instrumental activities of daily living," *Gerontologist*, vol. 9, no. 3, pp. 179–186, Sep. 1969.
[23] J. M. N. Ramírez and Y. Cardinale, "Activities of daily living detection on healthcare: A categorization," in *Proc. 7th Int. Workshop Sensor-Based Activity Recognit. Artif. Intell.*, Sep. 2022, pp. 1–7.
[24] N. Twomey, T. Diethe, X. Fafoutis, A. Elsts, R. McConville, P. Flach, and I. Craddock, "A comprehensive study of activity recognition using accelerometers," *Informatics*, vol. 5, no. 2, p. 27, May 2018.
[25] G. L. Santos, P. T. Endo, K. H. D. C. Monteiro, E. D. S. Rocha, I. Silva, and T. Lynn, "Accelerometer-based human fall detection using convolutional neural networks," *Sensors*, vol. 19, no. 7, p. 1644, Apr. 2019.
[26] P. Palimkar, V. Bajaj, A. K. Mal, R. N. Shaw, and A. Ghosh, "Unique action identifier by using magnetometer, accelerometer and gyroscope: KNN approach," in *Advanced Computing and Intelligent Technologies*. Berlin, Germany: Springer, 2022, pp. 607–631.
[27] C. Crema, A. Depari, A. Flammini, E. Sisinni, T. Haslwanter, and S. Salzmann, "Characterization of a wearable system for automatic supervision of fitness exercises," *Measurement*, vol. 147, Dec. 2019, Art. no. 106810.
[28] S. Paraschiakos, R. Cachucho, M. Moed, D. van Heemst, S. Mooijaart, E. P. Slagboom, A. Knobbe, and M. Beekman, "Activity recognition using wearable sensors for tracking the elderly," *User Model. User-Adapted Interact.*, vol. 30, no. 3, pp. 567–605, Jul. 2020.
[29] J. Medina-Quero, S. Zhang, C. Nugent, and M. Espinilla, "Ensemble classifier of long short-term memory with fuzzy temporal windows on binary sensors for activity recognition," *Exp. Syst. Appl.*, vol. 114, pp. 441–453, Dec. 2018.
[30] L. Chen, X. Liu, L. Peng, and M. Wu, "Deep learning based multimodal complex human activity recognition using wearable devices," *Int. J. Speech Technol.*, vol. 51, no. 6, pp. 4029–4042, Jun. 2021.
[31] Y. Hartmann, H. Liu, and T. Schultz, "Feature space reduction for human activity recognition based on multi-channel biosignals," in *Proc. 14th Int. Joint Conf. Biomed. Eng. Syst. Technol.*, 2021, pp. 215–222.
[32] H. Sfar and A. Bouzeghoub, "Activity recognition for anomalous situations detection," *Innov. Res. Biomed. Eng.*, vol. 39, no. 6, pp. 400–406, 2018.
[33] K. Xia, J. Huang, and H. Wang, "LSTM-CNN architecture for human activity recognition," *IEEE Access*, vol. 8, pp. 56855–56866, 2020.
[34] I. M. Pires, F. Hussain, G. Marques, and N. M. Garcia, "Comparison of machine learning techniques for the identification of human activities from inertial sensors available in a mobile device after the application of data imputation techniques," *Comput. Biol. Med.*, vol. 135, Aug. 2021, Art. no. 104638.
[35] S. Katz, T. D. Downs, H. R. Cash, and R. C. Grotz, "Progress in development of the index of ADL," *Gerontologist*, vol. 10, no. 1, pp. 20–30, Mar. 1970.
[36] C. Graf, "The Lawton instrumental activities of daily living scale," *AJN, Amer. J. Nursing*, vol. 108, no. 4, pp. 52–62, 2008.
[37] M.-E. Joél, S. Dufour-Kippelen, C. Duchěne, and M. Marmier. (2010). *The Long-term Care Syst. for Elderly France*. Centre for European Policy Studies. [Online]. Available: https://www.ceps.eu/ceps-publications/long-term-care-system-elderly-france/
[38] P. Parvin, S. Chessa, M. Manca, and F. Paterno, "Real-time anomaly detection in elderly behavior with the support of task models," *Proc. ACM Hum.-Comput. Interact.*, vol. 2, pp. 1–18, Jun. 2018.
[39] Y. Francillette, B. Bouchard, K. Bouchard, and S. Gaboury, "Modeling, learning, and simulating human activities of daily living with behavior trees," *Knowl. Inf. Syst.*, vol. 62, no. 10, pp. 3881–3910, Oct. 2020.
[40] S. Shao, N. Kubota, K. Hotta, and T. Sawayama, "Behavior estimation based on multiple vibration sensors for elderly monitoring systems," *J. Adv. Comput. Intell. Intell. Informat.*, vol. 25, no. 4, pp. 489–497, Jul. 2021.
[41] G. Bouaziz, D. Brulin, H. Pigot, and E. Campo, "Detection of social isolation based on meal-taking activity and mobility of elderly people living alone," *IRBM*, vol. 44, no. 4, Aug. 2023, Art. no. 100770.
[42] Y. Li, G. Yang, Z. Su, S. Li, and Y. Wang, "Human activity recognition based on multienvironment sensor data," *Inf. Fusion*, vol. 91, pp. 47–63, Mar. 2023.
[43] G. Giannios, L. Mpaltadoros, V. Alepopoulos, M. Grammatikopoulou, T. G. Stavropoulos, S. Nikolopoulos, I. Lazarou, M. Tsolaki, and I. Kompatsiaris, "A semantic framework to detect problems in activities of daily living monitored through smart home sensors," *Sensors*, vol. 24, no. 4, p. 1107, Feb. 2024.
[44] W. Qi, H. Su, and A. Aliverti, "A smartphone-based adaptive recognition and real-time monitoring system for human activities," *IEEE Trans. Hum.-Mach. Syst.*, vol. 50, no. 5, pp. 414–423, Oct. 2020.

[45] Y.-L. Hsu, S.-C. Yang, H.-C. Chang, and H.-C. Lai, "Human daily and sport activity recognition using a wearable inertial sensor network," *IEEE Access*, vol. 6, pp. 31715–31728, 2018.
[46] L. Peng, L. Chen, M. Wu, and G. Chen, "Complex activity recognition using acceleration, vital sign, and location data," *IEEE Trans. Mobile Comput.*, vol. 18, no. 7, pp. 1488–1498, Jul. 2019.
[47] V. R. Chifu, C. B. Pop, A. M. Rancea, A. Morar, T. Cioara, M. Antal, and I. Anghel, "Deep learning, mining, and collaborative clustering to identify flexible daily activities patterns," *Sensors*, vol. 22, no. 13, p. 4803, Jun. 2022.
[48] Y. Zhao, J. Liu, C. Lian, Y. Liu, X. Ren, J. Lou, M. Chen, and W. J. Li, "A single smart ring for monitoring 20 kinds of multi-intensity daily activities—From kitchen work to fierce exercises," *Adv. Intell. Syst.*, vol. 4, no. 12, Dec. 2022, Art. no. 2200204.
[49] G. A. Oguntala, Y. F. Hu, A. A. S. Alabdullah, R. A. Abd-Alhameed, M. Ali, and D. K. Luong, "Passive RFID module with LSTM recurrent neural network activity classification algorithm for ambient-assisted living," *IEEE Internet Things J.*, vol. 8, no. 13, pp. 10953–10962, Jul. 2021.
[50] U. Saeed, S. Yaseen Shah, S. Aziz Shah, H. Liu, A. Alhumaidi Alotaibi, T. Althobaiti, N. Ramzan, S. Ullah Jan, J. Ahmad, and Q. H. Abbasi, "Multiple participants' discrete activity recognition in a well-controlled environment using universal software radio peripheral wireless sensing," *Sensors*, vol. 22, no. 3, p. 809, Jan. 2022.
[51] L. Yuan, J. Andrews, H. Mu, A. Vakil, R. Ewing, E. Blasch, and J. Li, "Interpretable passive multi-modal sensor fusion for human identification and activity recognition," *Sensors*, vol. 22, no. 15, p. 5787, Aug. 2022.
[52] J. Howe, W. Chua, E. Sumner, B. Drozdowska, R. Laverick, R. L. Bevins, E. Jean-Baptiste, M. Russell, P. Rotshtein, and A. M. Wing, "The efficacy of a task model approach to ADL rehabilitation in stroke apraxia and action disorganisation syndrome: A randomised controlled trial," *PLoS ONE*, vol. 17, no. 3, Mar. 2022, Art. no. e0264678.
[53] F. Liang, Z. Su, and W. Sheng, "Multimodal monitoring of activities of daily living for elderly care," *IEEE Sensors J.*, vol. 24, no. 7, pp. 11459–11471, Apr. 2024.
[54] J. F. Allen, "Maintaining knowledge about temporal intervals," *Commun. ACM*, vol. 26, no. 11, pp. 832–843, Nov. 1983.
[55] D. Moldovan, A. Olosutean, V. Chifu, C. Pop, T. Cioara, I. Anghel, and I. Salomie, "Big data analytics for the daily living activities of the people with dementia," in *Proc. IEEE 14th Int. Conf. Intell. Comput. Commun. Process. (ICCP)*, Sep. 2018, pp. 175–181.
[56] G. Mokhtari, A. Anvari-Moghaddam, and Q. Zhang, "A new layered architecture for future big data-driven smart homes," *IEEE Access*, vol. 7, pp. 19002–19012, 2019.
[57] F. Xu and H. Wu, "Research on smart home care model under the background of big data," in *Proc. 2nd Int. Conf. Inf. Technol. Comput. Appl. (ITCA)*, Dec. 2020, pp. 546–550.
[58] R. Zhang, "Thinking and practice of smart home product design driven by big data," in *Proc. 3rd Int. Academic Exchange Conf. Sci. Technol. Innov. (IAECST)*, Dec. 2021, pp. 354–358.
[59] N. Vajagic and M. Antic, "Smart home IoT network diagnostics (don't short) using big data services," in *Proc. 30th Telecommun. Forum (TELFOR)*, Nov. 2022, pp. 1–4.
[60] A. Yassine, S. Singh, and A. Alamri, "Mining human activity patterns from smart home big data for health care applications," *IEEE Access*, vol. 5, pp. 13131–13141, 2017.
[61] S. Hurtado, J. García-Nieto, A. Popov, and I. Navas-Delgado, "Human activity recognition from sensorised patient's data in healthcare: A streaming deep learning-based approach," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 8, no. 1, p. 23, 2023.
[62] C. Martinie, P. Palanque, M. Ragosta, and R. Fahssi, "Extending procedural task models by systematic explicit integration of objects, knowledge and information," in *Proc. 31st Eur. Conf. Cognit. Ergonom.*, Aug. 2013, pp. 1–10.
[63] C. Martinie, P. Palanque, E. Barboni, and M. Ragosta, "Task-model based assessment of automation levels: Application to space ground segments," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2011, pp. 3267–3273.
[64] C. Martinie, D. Navarre, P. Palanque, and C. Fayollas, "A generic tool-supported framework for coupling task models and interactive applications," in *Proc. 7th ACM SIGCHI Symp. Eng. Interact. Comput. Syst.*, New York, NY, USA, Jun. 2015, pp. 244–253.
[65] J. Annett, "Hierarchical task analysis (HTA)," in *Handbook of Human Factors and Ergonomics Methods* (hta). Boca Raton, FL, USA: CRC Press, 2004, pp. 355–363.
[66] J. M. N. Ramírez, P. Roose, M. Dalmau, and M. Bakni, "Proposal and validation of a domaine specific language for the representation of the AGGIR constants," in *Proc. Int. Conf. Health Informat.*, 2018, pp. 438–445.
[67] P. Lalanda, C. Hamon, C. Escoffier, and T. Leveque, "ICasa, a development and simulation environment for pervasive home applications," in *Proc. IEEE 11th Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2014, pp. 1142–1143.
[68] Alliance OSGi. (2009). *OSGI Alliance: OSGI Service Platform, Core Specification, Release 4, Version 4.2*. [Online]. Available: https://docs.osgi.org/download/r4v42/r4.core.pdf
[69] F. Buckinx, E. Peyrusqué, M. J. Kergoat, and M. Aubertin-Leheudre, "Reference standard for the measurement of loss of autonomy and functional capacities in long-term care facilities," *J. Frailty Aging*, vol. 12, pp. 236–243, Jul. 2023.
[70] H. Mshali, T. Lemlouma, and D. Magoni, "Analysis of dependency evaluation models for eHealth services," in *Proc. IEEE Global Commun. Conf.*, Dec. 2014, pp. 2429–2435.
[71] R. B. Miller, "Response time in man-computer conversational transactions," in *Proc. Fall Joint Comput. Conf., AFIPS*, 1968, pp. 267–277.
[72] M. Stonebraker, U. Çetintemel, and S. Zdonik, "The 8 requirements of real-time stream processing," *ACM SIGMOD Rec.*, vol. 34, no. 4, pp. 42–47, Dec. 2005.
[73] M. Karatas, L. Eriskin, M. Deveci, D. Pamucar, and H. Garg, "Big data for healthcare Industry 4.0: Applications, challenges and future perspectives," *Exp. Syst. Appl.*, vol. 200, Aug. 2022, Art. no. 116912.
[74] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognit. Lett.*, vol. 119, pp. 3–11, Mar. 2019.
[75] D. Bouchabou, S. M. Nguyen, C. Lohr, B. LeDuc, and I. Kanellos, "A survey of human activity recognition in smart homes based on IoT sensors algorithms: Taxonomies, challenges, and opportunities with deep learning," *Sensors*, vol. 21, no. 18, p. 6037, Sep. 2021.
[76] F. Kulsoom, S. Narejo, Z. Mehmood, H. N. Chaudhry, A. Butt, and A. K. Bashir, "A review of machine learning-based human activity recognition for diverse applications," *Neural Comput. Appl.*, vol. 34, no. 21, pp. 18289–18324, Nov. 2022.
[77] S. P. Chatrati, G. Hossain, A. Goyal, A. Bhan, S. Bhattacharya, D. Gaurav, and S. M. Tiwari, "Smart home health monitoring system for predicting type 2 diabetes and hypertension," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 3, pp. 862–870, Mar. 2022.
[78] A. Yadav, N. Ahmad, I. R. Khan, P. Agarwal, and H. Kaur, "Role of AI, big data in smart healthcare system," in *Proc. 6th Int. Conf. Inf. Syst. Comput. Netw. (ISCON)*, Mar. 2023, pp. 1–8.
[79] A. L'Heureux, K. Grolinger, H. F. Elyamany, and M. A. M. Capretz, "Machine learning with big data: Challenges and approaches," *IEEE Access*, vol. 5, pp. 7776–7797, 2017.
[80] B. Qolomany, A. Al-Fuqaha, A. Gupta, D. Benhaddou, S. Alwajidi, J. Qadir, and A. C. Fong, "Leveraging machine learning and big data for smart buildings: A comprehensive survey," *IEEE Access*, vol. 7, pp. 90316–90356, 2019.
[81] J. K. Reena and R. Parameswari, "A smart health care monitor system in IoT based human activities of daily living: A review," in *Proc. Int. Conf. Mach. Learn., Big Data, Cloud Parallel Comput. (COMITCon)*, Feb. 2019, pp. 446–448.
[82] J. W. Lee, S. Cho, S. Liu, K. Cho, and S. Helal, "Persim 3D: Context-driven simulation and modeling of human activities in smart spaces," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 4, pp. 1243–1256, Oct. 2015.
[83] J. Lundström, J. Synnott, E. Järpe, and C. D. Nugent, "Smart home simulation using avatar control and probabilistic sampling," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops*, Mar. 2015, pp. 336–341.
[84] N. Alshammari, T. Alshammari, M. Sedky, J. Champion, and C. Bauer, "OpenSHS: Open smart home simulator," *Sensors*, vol. 17, no. 5, p. 1003, May 2017.
[85] T. Musharu and D. Vogts, "Extended sesim: A tool to support the generation of synthetic datasets for human activity recognition," in *Proc. Conf. e-Bus., e-Services e-Soc.* Cham, Switzerland: Springer, 2022, pp. 148–163.
[86] B. Park, H. Min, G. Bang, and I. Ko, "The user activity reasoning model in a virtual living space simulator," *Int. J. Softw. Eng. Appl.*, vol. 9, no. 6, pp. 53–62, Jun. 2015.

**YUDITH CARDINALE** received the degree (Hons.) in computer engineering from the Universidad Centro-Occidental Lisandro Alvarado, Venezuela, in 1990, and the M.Sc. and Ph.D. degrees in computer science from the Universidad Simón Bolívar (USB), Venezuela, in 1993 and 2004, respectively. Since 1996, she has been a Full Professor with the Computer Science Department, USB. Since 2018, she has been a Full Professor and a Researcher with Valencian International University, Spain, and a Principal Researcher of the Research Group in Data Science (GRID). She is currently an Associate Researcher with the Universidad Católica San Pablo, Arequipa, Perú. She has been the coordinator of a variety of international research projects. She has written a range of scientific articles published in international journals, books, and conferences. Her research interests include parallel processing, distributed systems, operating systems, big data and data science, high performance on grid and cloud platforms, collaborative frameworks, and web services, including semantic web. She has participated as a member of program committees of several international conferences and journals. She is the President of the Venezuelan Computer Society.

**JOSÉ MANUEL NEGRETE RAMÍREZ** received the Ph.D. degree in computer science from the Université de Pau et des Pays de l'Adour.

He is currently a Researcher in the field of computer science. His expertise spans a wide range of topics, including artificial intelligence, machine learning, model-driven languages, and smart health computer systems. Throughout his academic career, he has published numerous research articles in top-tier research journals and presented his work at renowned international conferences. As an Educator, he is dedicated to sharing his knowledge and experience with students. He has mentored many master's and Ph.D. students, guiding them in their research projects and helping them develop their computer science skills. His current work focuses on developing new techniques and technologies in computer science to solve complex problems and address emerging challenges in various application domains.

**IRACHE GARAMENDI BRAGADO** received the bachelor's degree in physical sciences from the Complutense University of Madrid, in 2008, and the master's degree in big data and data science, in 2022.

Since 2008, she has been working in the private sector, with a focus on the field of computer science. Her experience spans various domains, including software development in the banking sector, as well as quality assurance for the telecommunications and human resources software industries. Her goal is to continue exploring new technologies and applications that enhance efficiency and responsiveness in the field of software development and process automation. Her work focuses on software development, process automation, automatic deployment, and integration of generative AI.

**ISMAEL DE FEZ** received the telecommunications engineering and M.S. degrees in telematics and the Ph.D. degree in telecommunications from the Universitat Politècnica de València (UPV), Valencia, Spain, in 2007, 2010, and 2014, respectively. Since 2007, he has been a Researcher with the Multimedia Communications Research Group (COMM), Institute of Telecommunications and Multimedia Applications (iTEAM), UPV. Since 2021, he has been an Assistant Professor with Valencian International University (VIU). He is currently a Researcher with the Research Group in Data Science (GRID), VIU. His research interests include multimedia communications, streaming and file transmission, and the Internet of Things.