

Context-aware decision making under uncertainty for voice-based control of smart home

Pedro Chahuara, François Portet, Michel Vacher

▶ **To cite this version:**

Pedro Chahuara, François Portet, Michel Vacher. Context-aware decision making under uncertainty for voice-based control of smart home. Expert Systems with Applications, 2017, 75, pp.63-79. 10.1016/j.eswa.2017.01.014 . hal-01450843

HAL Id: hal-01450843
https://hal.science/hal-01450843

Submitted on 31 Jan 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Context-Aware Decision Making Under Uncertainty for Voice-based Control of Smart Home

PEDRO CHAHUARA, Univ. Grenoble Alpes, CNRS, LIG, F-38000 Grenoble France,
FRANÇOIS PORTET, Univ. Grenoble Alpes, CNRS, LIG, F-38000 Grenoble France,
MICHEL VACHER, Univ. Grenoble Alpes, CNRS, LIG, F-38000 Grenoble France
E-mail: {pedro.chahuara,francois.portet,michel.vacher}@imag.fr

ABSTRACT

This paper presents a framework to build home automation systems reactive to voice for improved comfort and autonomy at home. The focus of this paper is on the context-aware decision process which must reason from uncertain facts inferred from real sensor data. This framework for building context aware systems uses a hierarchical knowledge model so that different inference modules can communicate and reason with same concepts and relations. The context-aware decision module is based on a Markov Logic Network, a recent approach which make it possible to benefit from formal logical representation and to model uncertainty of this knowledge. In this work, uncertainty of the decision model has been learned from data. Although some expert systems are able to deal with uncertainty, the Markov Logic Network approach brings a unified theory for dealing with logical entailment, uncertainty and missing data. Moreover, the ability to use a priori knowledge and to learn weights and structure from data make this model appealing to address the challenge of adaptation of expert systems to new applications. Finally, the framework has been implemented in an on-line system which has been evaluated in a real smart home with real naive users. Results of the experiment show the interest of context-aware decision making and the advantages of a statistical relational model for the framework.

KEYWORDS: Keywords: Decision Making ; Voice User Interface ; Markov Logic Network ; Smart Home Ambient Assisted Living.

Reference of the published paper of this draft:

- **Article reference:** ESWA11062
- **Journal title:** Expert Systems With Applications
- **Accepted manuscript available online:** 20-JAN-2017
- **DOI information:** 10.1016/j.eswa.2017.01.014

1 Introduction

In the domain of Ambient Intelligence (AmI), Smart Homes (SH) have emerged to increase users' experience and control as well as to provide support to people with special needs in their daily life. To provide this enriched control, these systems perceive their environment and must decide which actions to apply to the environment. This decision can be made after some specific events such as a user's request, a schedule (e.g., programmed task) or a detected risky situation. This perception is not only useful to trigger a decision but also to adapt the decision to make to the current circumstances in which such decision must be performed. In the AmI domain, these circumstances are called *the context* and systems, that explicitly take the context into account, are said *context-aware*. According to (Dey, 2001), "context is any information that can be used to characterize the situation of an entity". The information of the environment that determine the behaviour of context aware systems depends on the purposes of each application, but in general, the time, the location of the user, the activity she performs, and her identity are common elements of the context. In this paper, we present a framework to handle such context in order to improve decision making in a smart home controlled by voice.

To illustrate the role of context in decision making, let's take the example of a voice based controlled smart home. These kinds of control are based on a Voice-User Interface (VUI) which becomes popular in the smart home domain (Istrate et al., 2008; Badii and Boudy, 2009; Hamill et al., 2009; Filho and Moir, 2010; Gemmeke et al., 2013; Cristoforetti et al., 2014) due to its mature technology and because it provides interaction using natural language (Portet et al., 2013; Vacher et al., 2015). Moreover, it is well adapted to people with reduced mobility and to some emergency situations (hands-free and distant interaction) because it does not force the user to be physically at a particular place in order to operate. In such settings, voice command can be 'turn on the light', 'check the door', 'call my daughter', etc. In that case, context is useful to disambiguate the order and to make relevant decision. In an utterance such as 'turn on the light' in a room with several lamps some of which being dimmed, the user does not tend to specify which lamps are to be activated and at which intensity (Vacher et al., 2015) because she would expect her interlocutor to guess these correctly. Since it would not be natural to ask the user to specify all details of an order, decision making systems must evaluate the context to make the most adequate decision. For instance, if the user is asking to turn on the light when tidying up the room, the ceiling light at full intensity might be the best decision as the sofa lamp would if the user was awaking from a nap. This simple example illustrates the need and role of context for disambiguation and natural interaction.

The analysis of the context is performed when the system recognizes a specific circumstance which has been specified a priori. Indeed, when designing the functionality of a smart home a list of situations, circumstances of interest, must be defined. These situations are related to the services we want to provide to the user and their definition depend mainly on chief goals of the project, the physical environment, and the preferences and profile of the inhabitants. For instance, as an objective to guarantee security, the system could be designed to detect a possible fire within the home, therefore the situation in this case is given by a set of conditions that can bring to light a fire: a sudden rise of temperature, a detection of smoke, etc. The context in the same example could be provided by activity of the inhabitant. Whether she is cooking or sleeping has a big influence on the evaluation of how risky the situation is and the system is expected to act according to this. Therefore, the situation specifies 'when' the system should act, whereas the context influences 'how' the system responds to the user's needs. Context and

situations representation appear as a challenge in pervasive environments since they consist on knowledge that should, on the one hand, be easily configured by and for the user to adapt to new preferences or new settings in the physical environment and, on the other hand, should be able to handle the uncertainty on the data and to automatically adapt to new situation in the data. They require the application of knowledge representation models featuring high readability, modularity and expressiveness.

When executing a command in response to a detected situation, the inference of an appropriate response might be based on uncertain and imprecise evidence that make unsuitable the application of simple reactive rules. In real applications, it frequently happens that it is only possible to have a degree of belief about the current state of the context because complex information, such as activity recognition, is inferred from indirect sources of data. Therefore, the treatment of contextual information requires to take in account a measure of belief. This is even more important when the response should be regulated according to the degree of uncertainty or when the uncertainty on the context is so high that there is a risk of executing a wrong command. Alternatives, risks and uncertainty are elements of decision making in real applications that must be treated in a formal framework to model their influence on the final decision. Nowadays, most of the methods applied for decision making in smart homes consist of classification algorithms which infer the right response given the evidence of the environment without modelling the decision process. Furthermore, due to the complexity and cost of experimentation in smart homes too few methods have been evaluated in realistic setting, leaving unknown the practical and user's experience aspects of these methods. Apart from the uncertainty handling, there is then a second main requirement in the development of smart homes, the design of decision making models that specifies formally the variables that take part in a decision process: how they are influenced by the evidence received from the environment, what are the relations among them , and what are the consequences of the final actions on the system.

To address the above challenges, we present, in this paper, the framework we implemented in the context of the SWEET-HOME project. The developed system uses the data streams of heterogeneous sensors installed in a real smart home to infer the current context. The context is inferred either when the inhabitant utters a vocal order or when a situation which can be risky for the user is detected. The system relies on an expressive knowledge representation and also includes elements to model decision process under uncertainty.

The main contributions of this work are:

- a framework for context aware systems using a hierarchical knowledge model that provides a common base so that different inference methods can share the same concepts and relations while allowing an efficient organization and interaction between the modules implementing these methods.
- a decision making method based on a logic model which can handle uncertainty and which can be learned from data. This method can be linked directly to the domain knowledge model and allows the inclusion of expert knowledge.
- an evaluation of the whole system *on-line*; including context-aware controller, speech recognition and home automation components; in a real environment with human participants performing daily life activities and interacting with the system.

The paper is organised as follows. After a brief overview of the state-of-the-art in Section 2,

Section 3 introduces the smart home considered in the study as well as the definition of the concepts that are necessary to define a context-aware voice based controller of smart homes. Section 4 details the architecture of such controller including the knowledge representation and the information inferred from raw data. The context aware decision making from heterogeneous data is then presented in Section 5. The controller has been evaluated in realistic conditions with naive users. These experiments and their results are reported in Section 6. The paper ends with a discussion of the approach and gives an outlook for further work.

2 Related Work

A common aspect among smart home projects in the literature is that generally the most important contextual elements are: the location of the user, the activity she performs, her identity and the time. This information is usually enough to analyse a situation under a specific condition. The research on context aware systems has recently been enriched by a vast amount of knowledge representation techniques in order to model the concepts related to this field. Logical approaches are generally applied for representation in pervasive environments because the facilities they provide to model relations among entities and also to perform automatic inference. Two families of approaches can be distinguished: logic programming and description logic.

Paper	Decision making	Context aware	Voice based	Uncertainty handling	knowledge representation	Online syst.	Smart home	Multi-room	End-user inclusion
(Ranganathan and Campbell, 2003; Loke, 2004)	not covered	Predefined situations	no	no	first order logic	no	no	yes	no experiment
(Katzouris et al., 2014)	not covered	activity	no	Probabilistic Event calculus	Logic based	no	no	—	synthetic data
(Mileo et al., 2010)	not covered	location, person, used object	no	no	Answer set programming	no	no	—	synthetic data
(Hamill et al., 2009)	action after 'Yes'/'No' answer	no	yes	no	predefined dialogue	yes	no	no	9 young speakers
(Liao and Tu, 2007)	action according to dangerous level	Predefined situations	no	probability density function	Ontology	no	no	no	synthetic data
(Katzouris et al., 2014)	not covered	activity	no	Probabilistic Event calculus	Logic based	no	no	—	synthetic data
(Gómez-Romero et al., 2012)	action according to situation recognition	activity predefined situations	no	no	ontology	yes	yes	no	Video sequence of one person
(Kofler et al., 2012)	actions predefined in ontology	predefined situations	no	no	ontology	no	no	—	no experiment
(Mitra et al., 2011)	influence diagram	user, device location	no	yes	Bayesian model	no	no	—	synthetic data
(Nishiyama et al., 2011)	influence diagram	predefined situations	no	yes	Bayesian model	no	no	—	synthetic data
(Helaoui et al., 2013)	not covered	activity	no	yes	Probabilistic ontology	yes	no	—	synthetic data
(Rodríguez et al., 2014)	not covered	activity	no	yes	Fuzzy ontology	yes	no	—	synthetic data
(Chahuara et al., 2013)	MLN based influence diagram	location, activity, agitation, predefined situations	yes	yes	OWL2 and MLN	yes	yes	yes	15 naive users
This study	MLN based influence diagram	location, activity, agitation, predefined situations	yes	yes	OWL2 and MLN	yes	yes	yes	37 naive users

Table 1: Summary of the studies related to context awareness in smart homes

With respect to the first method, (Ranganathan and Campbell, 2003) modelled actions to be executed in a pervasive environment by means of first order predicate calculus whereas (Loke, 2004) applied an extension of Prolog for representation and reasoning with complex situations which are defined as constraints on sensor readings such as a person occupying a certain room or a device being activated. (Mileo et al., 2010) also applied a logic programming approach, Answer Set Programming, in order to implement a system of prediction of risky situations. Event calculus, a method to model situations using first order logic, has also been applied in the development of Smart Homes (Chen et al., 2008; Katzouris et al., 2014). The main advantage of logic programming is that it allows very expressive human readable models.

However, currently the main trend for knowledge representation in smart environments is the use of ontologies based on Description Logic (DL). Even if DL has a reduced expressiveness compared with First Order Logic (FOL), it is an ideal framework to model taxonomies and relations among concepts. Contrary to logic programming, DL is based on the open world assumption, it means that no conclusion can be obtained about a fact that is not declared; thus every statement that is not declared remains possible. This feature is important to model incomplete information since it makes models easier to reuse: when the model must be adapted to a specific circumstance, the appropriate facts must be included. The Ontology Web Language (OWL) is the main implementation of DL, and it has been largely applied in pervasive environments (Wolf et al., 2008; Rodríguez et al., 2014; Liao and Tu, 2007). (Wolf et al., 2008) have employed the facilities ontologies provide for reusability in order to create the OpenAAL middleware that can be employed for context management in pervasive environments. Another work with ontologies was presented by (Liao and Tu, 2007) where temporal events are represented on RDF and the concepts about the smart home are organized in an ontology. Context-aware systems have also been developed using ontologies to model the current context for mobile applications (Attard et al., 2013; Yilmaz and Erdur, 2012). Fuzzy ontologies have received a lot of attention to be able to model imprecision and allow inference with vague concepts such as “hot weather”. (Rodríguez et al., 2014) have presented and application of fuzzy ontologies for activity recognition in pervasive environments. Another important contribution to make ontologies capable of dealing with uncertainty is the use of probabilistic models in ontology inference. A notable application of such approach has been presented by (Helaoui et al., 2013).

Besides context representation and situation detection, the second important aspect in a context aware system is the decision making process. Also in this part, several logical approaches are presented in the literature. (Moore et al., 2011) have developed a system that exploit a set of fuzzy rules in order to find the most appropriate action under a certain condition given by the context. Similarly, (Kofler et al., 2012) and (Gómez-Romero et al., 2012) used Description Logic to define the behaviour of a context-aware system which models context elements (activities) in an ontology. There are several works applying ECA (Event-Condition-Action) rules based systems into pervasive environments (Leong et al., 2009; Yau and Liu, 2006), however in these proposals the system is set *a priori* to execute an action given a specific configuration (the condition) and consequently the system does not adapt its behaviour to a degree of belief of the current situation, thus the concept of risk is not taken into account. Risk in decision making can be represented by the probability that a decision made can provoke negative effects, and then the risk is a direct consequence of the uncertainty. Therefore, it is necessary to rely on methods that can treat the uncertainty of the variables involved in the decision process. Bayesian networks, for instance, are among the most important methods for decision making with contextual information, as exemplified by the application presented by (Lacey and MacNamara, 2000) to improve the

mobility of blind people. (Lee and Cho, 2012) have modelled uncertainty on context-aware system by means of Bayesian networks but the method does non include formal elements of decision making such as risk and utility. Influence Diagrams (ID) is a method based on Bayesian networks that includes special variables to appropriately model the decision process. Some research works on context aware systems (Mitra et al., 2011; Nishiyama et al., 2011; Carolis and Cozzolongo, 2004) have relied on ID in order to model the decision process, treating the uncertainty and measuring the expected utility of possible actions. However, the expressiveness of this probabilistic approach is limited since only conditional dependencies among variables can be represented in the model, besides the fact that it is less human readable than logical approaches.

Table 1 lists the main attributes of the papers considered in this state of the art. It shows the variety of models that have been applied for decision making by contrast to the lack experimental real-time tests in real environments. Even when several approaches have been proposed to fill the gap between formal models of knowledge representation and inference with uncertain information, most of these works have been applied to a particular problem of context awareness, such as activity recognition, and did not consider the problem as a decision making one. In this work, we propose an approach we use ontologies to represent of concepts that comes into play during the decision and a set of logical rules. During the decision stage, these logic rules are used as template to construct an influence diagram based on Markov Logic Networks (MLN), a statistical method that makes probabilistic inference from a model consisting of weighted logic rules (Richardson and Domingos, 2006). The remaining of this paper presents this framework and its application to decision making in a smart home.

3 Decision making in a voice-based controlled Smart Home context

The typical smart homes considered in the study are the ones that permit voice based interaction. Voice-User Interface (VUI) in domestic environments has recently gained interest in the speech processing community as exemplified by the rising number of smart home projects that consider Automatic Speech Recognition (ASR) in their design (Istrate et al., 2008; Charalampos and Maglogiannis, 2008; Popescu et al., 2008; Badii and Boudy, 2009; Hamill et al., 2009; Filho and Moir, 2010; Lecouteux et al., 2011; Gemmeke et al., 2013; Christensen et al., 2013; Cristoforetti et al., 2014). This kind of interface is particularly adapted to people in loss of autonomy (Portet et al., 2013; Vacher et al., 2015). The smart homes we are considering contains multiple rooms which are fitted with sensors and actuators such as infra-red presence detectors, electric meter, Multimedia server, etc. These smart homes are aiming at providing daily living context-aware decision using the perception of the situation of the user. In a voice based control application, the smart home should be *reactive* to vocal or other commands to make the most adequate action based on context, and could act *pro-actively* by recognising a specific situation in which an action must be made (e.g., for security issue). The two examples below illustrate this support:

> **Scenario 1** *The inhabitant wakes up at night and utters the vocal order "Turn on the light". This simple command requires context information (location and activity) to realize which light to turn on and the most appropriate intensity. In this case, the system decides to turn on the bedside lamp with a middle intensity since the ceiling light could affect her eyes sensitivity at that moment.*

> **Scenario 2** *The inhabitant returns to her flat after shopping, forgets to lock the door, and does*

her usual activities until night. She prepares to sleep and turns all the lights off but the bedside lamp as she usually reads before sleeping. After some minutes, she turns off the lamp and, from the sequence of her interactions with the environment, the system recognizes that she is about to sleep. The unlocked main door represents a relatively risky situation.The system could send a message through a speech synthesizer – considering the risk of interrupting her rest– to remind her to close the door.

The scenarios emphasize the role of contextual information (here activity and location), to provide the most suited support to the user. We define Location and Activity as follows:

Definition 1 [Location] $l(t) \in L$, where L is the set of predefined locations in the SH and $t \in \mathbb{N}$ is the time, specifies where the inhabitant is located.

In this work, a specific area corresponds to a room and we assume a single inhabitant in the environment.

Definition 2 [Activity] Routine activities performed during daily life; such as, sleeping, cooking, or cleaning. The set of activities is $\mathscr{A}$. At an instant t, the activity might be undetermined; so an activity occurrence is defined in an interval of time $[t_b; t_e]$, $A =< a, t_b, t_e >$, considering $a \in \mathscr{A}, (t_b, t_e) \in \mathbb{N}^2$ and $t_b < t_e$

Furthermore, a larger set of information can be extracted from the raw data. For instance, agitation, communication, etc. We call these *sources of information*:

Definition 3 [Source of Information] The system contains a set of variables V that describe the environment. A source of information is a variable $V_i \in V$ with domain $Dom(V_i)$ representing the information provided by a sensor or an inference process i.

Definition 4 [System state] If Υ is the set of possible values of V, a system state is an assignment $v \in \Upsilon$ making $V = \{V_1 = v_1, V_2 = v_2, ..., V_n = v_n\}$

The Situation is then defined by:

Definition 5 [Situation] A situation $S \subset \Upsilon$ is defined by a set of constraints $C = \{C_1^{k_1}, C_2^{k_2}, ..., C_m^{k_m}\}$, where each constraint $C_i^{k_i}$ establishes a set $D_i \subset Dom(V_{k_i})$ to constraint the value of a source of information V_{k_i}. Thus, $S = \{v / \forall C_i^{k_i} \in C, v_{k_i} \in D_i\}$

For example, in Scenario 2 presented before V_1, V_2 and $V_3, \ldots, V_n$ represent the state of the main door, the user's location and the state of the blinds and lights. A situation can be defined by constraints, $C_1^1, C_2^2, \ldots, C_n^n$, holding the following sets: $D_1 = \{open\}, D_2 = \{\neg kitchen\}, D_3 = \{off\}, \ldots, D_n = \{off\}$. A situation is recognized when all the lights are off, the blinds are closed, the front door is open and the person is not in the kitchen (assuming the front door is in the kitchen).

Definition 6 [Temporal Situation] A temporal situation R is defined by a set of constraints $T = \{T_1, T_2, ..., T_m\}$, where each T_k is a tuple composed of a pair of situations (S_k^1, S_k^2) and a temporal constraint r_k between S_k^1 and S_k^2, $T_k =< S_k^1, S_k^2, r_k >$.

Consider $T_l =< S^1, S^2, r >$ with $r = [t_i, t_j]$, a temporal situation is recognized when $t_i \leqslant t_S^2 - t_S^1 \leqslant t_j$ where t_S^i is the occurrence time of S^i. r_k can also be qualitative constraint such as $after(S_1, S_2)$ or $order(S_1, S_2, S_3)$. For more details about temporal representation and reasoning the reader is referred to (Artikis et al., 2012). In the rest of the paper we refer to temporal situations simply as situations.

The elements defined above compose the context that we define as follows:

> **Definition 7** [Context] Set of information characterizing the circumstance under which an inference is made.

This definition is very close to the one given by Dey (Dey, 2001), "any information that can be used to characterize the situation of an entity". Context is generally used for disambiguation. Once a situation is recognized, several decisions can be taken with different effects. The context provides complementary information to evaluate the circumstance in terms of risk and utility (safety, efficiency, comfort...). These two notions are defined below:

> **Definition 8** [Risk] The risk is the probabilistic measure that a given action would have a negative outcome in the situation under consideration.

Although the definition of risk varies according to the domain of application, in decision making, risk is often a consequence of uncertainty which is evaluated by listing all the possible outcomes with their probability of occurrence and their consequences.

> **Definition 9** [Utility] The utility $U \in [0, 1]$ is the degree of preferences of a system state caused by applying an action.

When uncertainty arise, an action is likely to cause several effects. If the effect leads to a negative outcome, then the utility U takes a negative value. Hence, risk and utility have a relationship. To compute the risk for a given action, the probability of all the unwanted states (i.e., those with a negative value) must be computed.

For example, in Scenario 1, the decision making is triggered by the recognition of the voice command *"turn on the light"*. The context is the location (bedroom), the time (middle of the night) and the activity (sleeping). The action to make could be to switch on the bedside lamp or the ceiling light or both. The effects could be to decrease or increase comfort. Thus,, the risk of each action is given by its probability of having an unwanted effect (here, decrease comfort). The utility is the numerical value associated to each effect.

This illustrates that the action to be taken depends on the context. For instance, in Scenario 1, since the user just awake in the night, it is better to light the bedside lamp to avoid dazzling, but the ceiling lamp could be a better choice in a different context (e.g., when tidying up).

From a preliminary user study (Portet et al., 2013), it appeared that the main aspects for context-aware decision are the location of the inhabitant, the current activity and the period of the day. These informations are useful to eliminate ambiguity in the decision making process. Many studies have also considered location and activity as fundamental for context-aware inference (Schilit et al., 1994; Mileo et al., 2011; Blachon et al., 2014; Vacher et al., 2015).

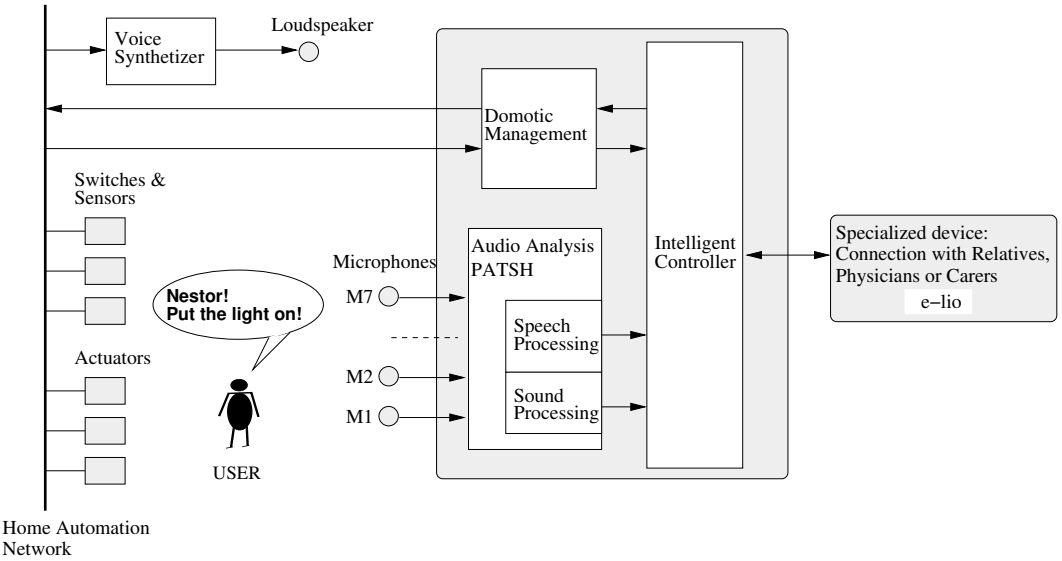


Figure 1: The SWEET-HOME system: example of a user asking to put on the light.

4 Intelligent Controller of the SWEET-HOME system

To provide assistance at home through voice command, the SWEET-HOME system is composed of an audio analysis system, called PATSH and an Intelligent Controller. The SWEET-HOME system is depicted in Figure 1. It is linked with an home automation network composed of typical home automation data sensors and actuators (switches, lights, blinds, etc.) and multimedia control (uPnP). Moreover, the system relies on several microphones per room disseminated into the ceiling so that voice command is made possible from anywhere in the house in a hands-free way. Finally, the system should also make easier for the user to connect with her relatives, physician or caregiver by using a dedicated communication system. In the case of SWEET-HOME this system is *e-lio* developed by the Technosens[1] company providing home services to the elderly people through the *e-lio* box (e.g., video-conferencing, calendar, photos, etc.).

In order for the user to be in full control of the system and also in order to adapt to the users' preferences, two ways of commanding the system are possible: voice command or classical tactile interfaces (i.e., switches). Briefly, the functioning of the system is the following. All sensor data are acquired continuously by the system in an on-line manner[2]. Amongst the raw information, the raw audio data are analysed by PATSH and then speech as well as sound events are transmitted to the subsequent stages. The Intelligent Controller analyses continuously the streams of data and makes decision based on these. If a vocal command is detected and, according to the context (e.g., user's location), an home automation command is generated to make the light up, close the curtains or emit a warning message thanks to a voice synthesizer. The rest of this section details the general architecture of the controller, the knowledge representation as well as the different methods implemented for inference of contextual information. The decision making *per*

[1] `http://www.technosens.fr/`

[2] Here, on-line means that each time a new event appears, it is immediately queued for processing.

se is detailed in Section 5.

4.1 Architecture

The architecture of the intelligent controller is depicted in Figure 2. At the bottom of the figure external systems are connected to the controller to gather streams of data and send orders to the home automation system. The microphone data is processed by an audio processing chain delivering hypotheses about the sound or the sentences being uttered by the user following the method described in (Sehili et al., 2012). All these streams of information are captured and interpreted to recognize situations and to make decisions.

The estimation of the current situation is carried out through the collaboration of several processors, each one being specialized in the treatment of a source of information. All processors share the knowledge specified in two ontologies and use the same repository of facts. Furthermore, the access to the knowledge base is executed under a service oriented approach that allows any processor being registered to be notified only about particular events and to make inferred information available to other processors. This data and knowledge centred approach ensures that all the processors are using the same data structure and that the meaning of each piece of information is clearly defined among all of them.

The intelligent controller performs inference in several stages, from raw input data until the evaluation of situations. Each event is produced by the arrival of sensor information. These events are considered of low level as they do not require inference. Once they are stored in the fact base, processing modules are executed sequentially (e.g., location, then activity, then situation). Thus, each inference corresponding to a high level event is stored in the database and used subsequently by the next module. Within the controller architecture, other inference modules can be added without compromising the processing of the other components.

The following sections describe the knowledge representation and the inference modules.

4.2 Two-level knowledge representation

Ontologies allow the formal representation of knowledge on a domain by means of taxonomies of concepts and relations among them. They are used for instance in the implementation of information systems to define shared data schemas among services, and also to create high level queries. The knowledge base of the system was built using the Ontology Web Language 2 (OWL2) (OWL2, 2012) which is a set of mark-up languages based on Description Logic (DL) (Baader et al., 2008). Description logics is a family of formal knowledge representation derived from first order logic aiming at preserving decidability, consistency and satisfiability.

The knowledge of the controller is defined using two semantic layers: the *low-level* and the *high-level* ontologies. The two ontologies were implemented using Protégé, not only for domain knowledge representation, but also for storing the events resulting from the processing modules. Furthermore, situations were defined within the ontologies allowing description logic reasoners to evaluate if a situation is happening (see Section 4.4). Consequently, the importance of the ontology goes beyond the mere description of the environment.

The *low-level* ontology is devoted to the representation of raw data and network information

description. State, location, value and URI of switches and actuators are examples of element to be managed at this level. The *high-level* ontology represents concepts being used at the reasoning level. These concepts are organized in 3 main branches: the Abstract Entity, the Physical Entity, and the Event concept that represents the transient observations of one abstract entity involving zero or several physical entities (e.g., at 12:03 the dweller is sleeping). Instances in the *high-level* ontology are produced by the inference modules (e.g. activity, location, and situations) after treating information coming from sensors. This separation between low and high levels make it possible a higher re-usability of the reasoning layer when the sensor network and the home must be adapted (Klein et al., 2007).

Figure 3 shows some concepts and relations of both ontologies. The ABoxes serve as fact bases. Let's refer to the Scenario 2 when the inhabitant turns the bedside lamp off to sleep. The controller updates devices states in the low level ontology and it can be inferred, still at a low level, that every light in the bedroom is off (`current_state(switch1, off)`, `current_state(switch2, off)`). The knowledge base can relate the `switch2` state to the beside lamp (`has_application(bedLamp, switch2)`) which itself is located in the bedroom (`is_located(bedLamp, br1)`, `Bedroom(br1)`). In the high level ontology, the interaction with the switch lamp (`switch2`) is stored as a device event (`Device(d_evt1)`) having time and room as properties. At this stage, the module in charge of location is requested and it returns bedroom as the present location of the inhabitant since the last events, corresponding to switch deactivation, took place in this room (`has_location(l_evt1, bedroom)`). Then, the evidence of the inhabitant being in the bedroom, having all lights turned off, and the evening as the period of the day, can be used to infer that the current activity is sleeping (`has_activity(a_evt1, sleep)`). Finally, these inferences provide the context on which situation recognition is applied. Under the same scenario, if the inhabitant forgot to close the main door and a situation was defined for this case, the situation would be labelled as detected in the ontology. Detected situations are treated by the decision module explained in Section 5.

4.3 Raw and inferred information

The raw data captured within the smart home contains information that must be extracted for enhanced decision making. Four types of abstraction are considered: localisation of the inhabitant, classification of audio input into speech and sound (i.e., non-speech) events, the agitation level, and the activity performed by the inhabitant at a certain moment. The system relies on different methods to infer these high level information which are fired sequentially since the activity modules requires the other three information as shown in Figure 4.

4.3.1 Speech/sound detection

In this approach, sound events are detected in real time by the PATSH system which is external to the Intelligent Controller (Vacher et al., 2013). It is composed of several processing modules organised into a pipeline that perform the following tasks: (1) multichannel **Data acquisition** (16bits, 16kHz, 7 channels), (2) **Sound detection**, (3) **Sound/speech discrimination**, (4) **Sound classification**, (5) **Automatic Speech Recognition** (ASR) and extraction of voice commands, and (6) **Presentation**, communicating the sound event to the Intelligent Controller.

Briefly, the audio events are detected in real-time by an adaptive threshold algorithm based on a wavelet analysis and a SNR (Signal-to-Noise Ratio) estimation. The events are then classified into speech or everyday life sound by Gaussian Mixture Models (GMMs). The microphones being omnidirectional, a sound can be recorded simultaneously by multiple microphones; PATSH identifies these simultaneous events. In addition, to recognize only vocal orders and not all sentences uttered in the home, all sound events shorter than 150 ms and longer than 2.2 seconds were discarded as well as those whose SNR was below 0 dB. These values were chosen after a statistical study on a dataset (Vacher et al., 2011). Once the sound event is classified as speech (respectively non-speech sound), the sound object is send to the **ASR** module to extract voice commands (respectively to the sound classification module). Due to the focus of the study on the decision making, the sound classification and the Automatic Speech Recognition (ASR) systems will not be detailed in the paper (see (Sehili et al., 2012) for details). The last step of the ASR was composed of a voice command recognizer. Briefly, the best of the ASR output hypotheses was phonetised and a distance was computed against all the possible phonetised sentences of the grammar. For each comparison, the minimal Levenstein distance $\gamma(i, j)$ was computed using Dynamic Time Warping (DTW). If $\gamma(i, j)$ was above a certain threshold then a voice command was detected otherwise it was rejected. This approach permits to recover some decoding errors such as word declination or light variations (the light, the lighting, etc.). In a lot of cases, a miss-decoded word is orthographically close to the good one (due to the close pronunciation). Possible voice orders were defined using a very simple grammar as shown on Figure 5. Every command starts with a unique keyword that permits the system to know whether the person is talking to the smart home or not. In the following, we will use *'Nestor'* as keyword. The grammar was built after a user study that showed that targeted users prefer precise short sentences over more natural long sentences (Portet et al., 2013). In this study, although most of the seniors spontaneously controlled the home by uttering sentences, the majority said they wanted to control the home using keywords.

Once the ASR or the sound classification is performed, the **Presentation** module translates the sound objects into XML representation containing: the type of sound (speech or non-speech), the class of sound or the transcription of the speech, the SNR, the duration and the time of occurrence as well as the microphone source. This XML representation is sent to the intelligent controller through a SOAP connection.

4.3.2 Agitation

The agitation is a measure of the frequency of actions of the person in the house. This makes it possible to distinguish from activities that require many actions (e.g., cleaning the house) from quieter ones (e.g., sleeping). In our approach, the agitation is estimated not only from infra-red sensors but also from sound information as well as door contacts. It is obtaining by summing the number of sensor events (sound, speech, opening of doors, infra-red firing) produced during the minute preceding the vocal order.

4.3.3 Localisation

In smart homes, cheap localisation can be performed by using infra-red sensors detecting movement but these sensors can lack sensitivity. To improve localisation, our approach fuses information

coming from different data sources namely, infra-red sensors, door contacts and microphones. The data fusion model is composed of a two-level dynamic network (Chahuara et al., 2011) whose nodes represent the different location hypotheses and whose edges represent the strength (i.e., certainty) of the relation between nodes.

4.3.4 Activity recognition

This is one of the most important elements of context in pervasive environments and the most challenging. We defined previously a set of activities of daily living that are representative of the routine of the inhabitant. Afterwards, a probabilistic method based on Markov Logic Networks (MLN) was implemented to recognize the performed activity using sensor information and the other inferred information (Chahuara et al., 2016). Initially, sensor data is organised in temporal windows that are used to generate, in a following step, vectors of special attributes (e.g. percentage of time in each room, sounds produced within the window). At the end, vectors become the input of the classifier. The last step consists on the evaluation of every situation definition, according to previous inference results, to know if a situation of interest takes place or not. Markov Logic has been proposed by (Helaoui et al., 2011) to perform activity recognition when several activities are performed simultaneously. This approach could also be integrated in the framework presented in our work.

4.4 Situation Recognition

As situations are defined as temporal patterns of the system state, ontologies provide an appropriate foundation for situation recognition since the facts (i.e., the system state) can be stored as individuals into the ontology as well as a complete semantic description of the environment as well. Furthermore, temporal representation can be achieved by means of role properties among event concepts defining temporal relations such as *previous* and *next* which, through chaining property of OWL2, can generate the *after* and *before* relations. Under some restrictions, Datalogs describing situations as logic rules can be expressed with description logic and stored into ontologies (Hitzler and Parsia, 2009). This can be achieved by means of Semantic Web Rule Language (SWRL) which is a logic rules language for the semantic web resulting from the combination of OWL and RuleML (Rule Markup Language). However, the scope of this approach is very limited as it does not allow to specify complex definitions. But, even when it is constrained to safe rules (rules that can only be applied to instances explicitly declared in the ontology), it overcomes several limitations of DL while having the definitions still as part of the ontology. In OWL-DL it is no possible, for example, to define a relation as a composition of two other relations, however in SWRL extends OWL-DL in order to make such a definition possible. In addition, SWRL built-in functions further extend the semantics of context definitions.

For instance, the situation in which a person is leaving her house without having closed the windows can be described by the SWRL Rule 1. In this rule the predicate $has_associated_object$ relates a device event $?d$ with a specific object (here $door$), the rule body declares that two conditions must be fulfilled for the situation to be recognized: the door of the kitchen (main door of the flat) is open
($takes_place_in(?d, kitchen), state_value(?d, open)$), while at least one ($swlrb : moreThan(sqwrl : count(?w), 0)$) of the windows $?w$ in the bedroom is open

($Window(?w), located_in(?w, bedroom)$). $swrlb$ is the name space for the built-ins predicates of SWRL while $sqwrl$ are name space for the built-ins of the Semantic Query-Enhanced Web Rule Language (SQWRL) a SWRL-based language for querying OWL ontologies implemented in Protégé. In the last part of the body the predicate $has_application$ relates an object in the flat, in this case the window, with a corresponding sensor (variable $?a$). If all the premises are satisfied, then a
$BedroomWindowsOpen$ is detected.

Rule 1

DeviceEvent(?d), has_associated_object(?d, door), takes_place_in(?d, kitchen),
state_value(?d, open), Window(?w), located_in(?w, bedroom),Application(?a),
has_application(?w,?a),current_state(?a,open), swlrb:moreThan(sqwrl:count(?w), 0)
→ current_state(BedroomWindowsOpen, detected)

Following similar principle, the situation in scenario 2 can be modelled by Rule 2.

Rule 2

DeviceEvent(?l), has_associated_object(?l, light), takes_place_in(?l, bedroom),
state_value(?l, off), Window(?w), located_in(?w, bedroom),Application(?a1),
has_application(?w,?a1),current_state(?a1,open), swlrb:equals(sqwrl:count(?w), 0), Blind(?b),
located_in(?b, bedroom),Application(?a2), has_application(?b,?a2),current_state(?a2,open),
swlrb:equals(sqwrl:count(?b), 0), Door(?d), located_in(?d, kitchen),Application(?a3),
has_application(?b,?a3),current_state(?a3,open), swlrb:equals(sqwrl:count(?d), 1)
→ current_state(MainDoorOpen, detected)

5 Decision Making using Markov Logic Network

The decision making module is the main component of the intelligent controller. When a situation is recognized, this module employs the high level knowledge in order to construct dynamically a decision model that takes into account the context and its degree of uncertainty.

For instance, When the user asks to turn on the light "Nestor turn on the light", it does not specify the desired level of illumination (i.e., strong or soft) or which lamp to be lit (e.g., bedside or ceiling lamp). For the user, this information is implicit and indeed another human being would probably infer correctly the user's goal. But for the system, this lack of information must be recovered from its knowledge of the context: the controller must infer the location of the participant to turn on the light in the room where the participant stands and it must infer her activity to determine the appropriate illumination or lamp in the room (e.g., if she is asleep then the bedside lamp might be more appropriate than the ceiling one). In this example, the context is made of two inferred parameters from sensor data: the location and the activity.

In SWEET-HOME, the actions that the intelligent controller can make are the following:

- turn on/off the {light, radio}

- close/open the {blinds, curtains}
- give the {temperature, time}
- warn about {open windows, unlocked door}
- order the *e-lio* system to call a specific number or to make a emergency call.

These actions constitute a subset of a larger set of possible actions resulting from a user study (Portet et al., 2013). This study showed that the user (in that case, elderly people) were more interested by actions providing more security and avoiding dangerous manipulations. Actions saving time or related to food were not liked by the participants. For instance, automation of coffee machine was unanimously disliked given that it is an activity they want to keep actively doing as they have time to do so and it is part of a social activity (making coffee for their visitors). Of course, this set of actions must be adapted to any user and home, but this predefined list was useful for the evaluation of the system.

In this section we briefly describe the decision problem by influence diagram models and how it has been modelled by Markov Logic Network.

5.1 Modelling the decision making by Influence Diagrams

Influence diagrams (Howard and Matheson, 1981) are probabilistic models used to represent decision problems. They extend Bayesian networks – composed only of state nodes – by the inclusion of two types of node: action and utility. An action node is a variable corresponding to a decision choice (e.g., turning the light on or warning the user). The state nodes represent the variables in the problem domain that are affected by the actions. Utility nodes (see Definition 9) are variables that represent the utility value obtained as consequence of applying the decided actions. For instance, turning the light on at full intensity when the person is asleep would have a negative utility.

Formally, given a set of actions $\mathscr{A}$ and an assignment of choices $a \in \mathscr{A}$, the expected utility EU for a is computed by:

$$EU(a) = \sum_{X} P(X|a,e)U(X) \tag{1}$$

where X is a set of state nodes, $U(X)$ is the utility value of X and e is the evidence (e.g., the context). The process of finding the "best" decision consists of solving the Maximum Expected Utility (MEU) problem which demands to compute the EU of every possible assignment of a.

$$a_{best} = \operatorname*{argmax}_{a} EU(a) \tag{2}$$

Figure 6 shows an example of Influence Diagram, based on the Scenario 1. In this case, the setting of action variables, represented by rectangular nodes, indicates which *lights* are operated and their *intensity*. Oval nodes are the state nodes, some of which are affected by the decision, while the others belong to the context (within the dashed area). Two variables influence directly the utility: the *comfort* of the inhabitant and the suitability of the activated *lights location* that ideally should be the same as the inhabitant. Note that this location is not easy to determine

in some cases since the inhabitant could be moving in the smart home while uttering the vocal order.

The interest of influence diagrams is essentially its ability to easily represent the structure of a decision problem and the dependencies between variables. However, it is limited to propositional variables while a decision model could benefit from relational knowledge (e.g., turning on a light *next to* the room of the dweller). Yet, first order rules, though very expressive, cannot make it possible for an expert to express uncertainty. To overcome these limitations, we propose to model the decision process by a Markov Logic Network.

5.2 Markov Logic Networks (MLN)

MLN (Richardson and Domingos, 2006) combines first-order logic and Markov Networks, an undirected probabilistic graphical model. A MLN is composed of a set of first-order formulas each one associated to a weight that expresses a degree of truth. This approach softens the assumption that a logic formula can only be true or false. A formula in which each variable is replaced by a constant is *ground* and if it consists of a single predicate it is a *ground atom*. A set of ground atoms is a *possible world*. All possible worlds in a MLN are true with a certain probability which depends on the number of formulas they agree with and the weights of these formulas.

Let's consider F a set of first-order logic formulas, i.e. a knowledge base, $w_i \in \mathbf{R}$ the weight of the formula $f_i \in F$, and C a set of constants (in our case, input data). During the inference process (Richardson and Domingos, 2006), the MLN predicates are grounded and a Markov network $M_{F,C}$ is constructed where each random variable corresponds to a ground atom. The probability of a possible world $P(X = x)$ can then be estimated using Equation 3.

$$P(X = x) = \frac{1}{Z} exp\left(\sum_{f_i \in F} w_i n_i(x)\right) \tag{3}$$

where $Z = \sum_{x' \in \chi} exp\left(\sum_{f_i \in F} w_i n_i(x')\right)$ is a normalisation factor, χ the set of possible worlds, and $n_i(x)$ is the number of true groundings of the i^{th} clause in the possible world x.

Because computing Z involves grounding the whole network in each possible world, exact inference in MLN is intractable in most cases, so Markov Chain Monte Carlo methods are applied (Richardson and Domingos, 2006).

MLN models acquisition consists in two independent tasks: structure learning and weight learning. Structure can be obtained by applying machine learning methods, such as Inductive Logic Programming, or rules written by human experts. Weight learning is an optimisation problem that requires learning data. Weight learning can be achieve by maximizing the likelihood wrt a learning set x. If the ith formula is satisfied $n_i(x)$ times in x, then by using equation (3), the derivative of the log-likelihood wrt the weight w_i is given by equation (4).

$$\frac{\partial}{\partial w_i} \log P_w(X = x) = n_i(x) - \sum_{x'} P_w(X = x') n_i(x) \tag{4}$$

Where x' is a possible world in x. The sum is thus performed over all the possible worlds x' and $P_w(X = x')$ is $P(X = x')$ computed using the vector $w = (w_1, \ldots, w_i, \ldots)$. The maximisation of the likelihood is performed by an iterative process converging towards an optimal w. Unfortunately,

computing equation (4) is intractable in most cases. Thus, approximation method are used in practice such as the *Scaled Conjugate Gradient* (Lowd and Domingos, 2007).

Since a Markov network is more general than a Bayesian network, Influence diagrams can also be implemented by means of MLN (Nath and Domingos, 2009). Nath et al. (Nath and Domingos, 2010) have proposed an algorithm that evaluates all the choices in a set of actions without executing the whole inference process for each choice resulting in an efficient way to estimate the optimal assignment. We have considered this approach suitable for implementing decision making in our framework for two main reasons. Firstly, MLNs are logical rules which can be stored in an ontological representation, using domain concepts in order to keep a standard vocabulary besides achieving decision model readability. Secondly, it deals with the uncertainty related to context variables.

A MLN for the influence diagram of Figure 6 can be defined as follows:

Predicates	**Domain**	**Type**
Intensity	{low,high}	Action
LightLocation	{bedroom,kitchen,toilet...}	Action
Comfort	{low,medium,high}	Utility
RightArea	{good,bad,acceptable}	Utility
Location	{bedroom,kitchen,toilet...}	State
Activity	{sleep,eat,clean,dress...}	State
Agitation	{low,medium,high}	State

Rules

3.35	$LightLocation(l) \wedge Location(l) \rightarrow RightArea(good)$
0.12	$LightLocation(l1) \wedge Location(l2) \wedge NextTo(l1,l2) \rightarrow RightArea(acceptable)$
2.44	$LightLocation(l1) \wedge Location(l2) \wedge l1! = l2 \rightarrow RightArea(low)$
1.46	$Activity(+a) \wedge Agitation(+d1) \wedge Intensity(+d2) \rightarrow Comfort(high)$
-0.07	$Activity(+a) \wedge Agitation(+d1) \wedge Intensity(+d2) \rightarrow Comfort(medium)$
-0.2	$Activity(+a) \wedge Agitation(+d1) \wedge Intensity(+d2) \rightarrow Comfort(low)$

Utility Values

U(RightArea(bad))=-2 U(RightArea(fair))=0 U(RightArea(good))=2
U(Comfort(low))=-3 U(Comfort(medium))=0 U(Comfort(high))=3

Facts

NextTo(kitchen,bedroom) NextTo(bedroom,study)

In this model, the top part presents the different predicates or arity 1 and the domain of their term variable. The middle part shows the rules as well as the utility values that model the influence diagram. The bottom part exhibits an extract of the domain knowledge (here, the information about the smart home topology). Each rule is weighted so that uncertainty about this knowledge is taken into account. To illustrate this take for instance the following rules from the model:

...
3.35 $LightLocation(l1) \wedge Location(l1) \rightarrow RightArea(good)$
0.12 $LightLocation(l1) \wedge Location(l2) \wedge NextTo(l1,l2) \rightarrow RightArea(acceptable)$
...

The first rule expresses that if the chosen action is to light on the lamp located in $l1$ and that the user is also in location $l1$ then the state variable $RightArea$ takes the value $good$.

The number 3.35 translates the fact that the rule is very often true. The second rule states that if the chosen lamp is close to the location of the inhabitant, then it is an acceptable choice. However, its weight is low 0.12 meaning that a configuration of variables satisfying this rule is not very probable. In MLN the influence diagram is a set of weighed formulae. The weights are learned from data and are relative to each other. In the case of the example, the first rule is true a majority of the time in the training dataset while the second rule is true only for a few cases.

In the MLN model above the inference process requires some explicit knowledge about the location of the rooms that is express by the $NextTo$ predicate with no weights, what declares that they are always true (facts). The + operator before a variable indicates that all possible values of the variable will be instantiated, thus in the fourth rule, the variables of the predicates $activity, agitation, intensity$ have respectively $6, 3, 2$ constants, so this rule will give place to 36 instantiations with constant values. The model combines explicitly two important aspects of decision making in a context aware system: first, the influence of contextual variables and decisions made on our variable of interest ($Comfort, RighArea$); and second, a numerical assessment of how reliable is the rule that has been learned. If, for example, the person is in cleaning in the kitchen when uttering the command to turn on the light and the system decides to activate one in the bedroom, we can note from the model that even if the intensity of the light is high, the final utility will be penalised by having activated the light of a wrong location. Indeed, this mistake would have a much higher importance in the third rule than in the second one because of the difference in weights.

This MLN is a template for constructing a Markov network modelling an influence diagram. The evidential information required to perform decision inference is composed by the contextual elements: location, agitation, and activity. As shown Figure 7, once the decision module is triggered, it gets the evidence from the ontology instances that are used to ground the MLN and it generates an influence diagram (actually a Markov network). This grounded network is then used to compute the action that maximizes the expected utility using Equation 5 where *RA* means $RightArea$ and $LL\ LightLocation$.

$$EU(a) = \sum_{x \in \{bad, fair, good\}} P(RA(x) \mid a).U(RA(x)) + \sum_{x \in \{low, med., high\}} P(LL(x) \mid a).U(LL(x)) \tag{5}$$

Decision models are available for each kind of action that are related to an object that may exist at different places. For instance, when a voice command such as 'turn on the light' is received, then the decision about the lighting is executed, when it is about the blinds the decision model about the blinds is run and so on.

5.3 Making decision with uncertain evidence

Most decision making processes are designed to use *regular* (also called *hard*) evidence resulting in an observation a of a random variable X, so that it is assumed that the assignment $X = a$ is true after the observation. However, in some cases the source of the evidence cannot be

completely reliable making the evidence uncertain (Pan et al., 2006). As presented in Section 4.3, contextual information is inferred from the sensor data by means of different statistical methods. As such, this information is often uncertain and we assume such inferences to be provided with a probability measure. These uncertain results are the input evidence of the decision model. But, in the decision model, the expected utility is computed without taking the uncertainty of the evidence into account. For instance, if the activity recognition module gives the following activities with their probabilities: sleeping (.33), tidying up (.34) and resting (.33), the decision module will consider only the most probable activity and will possibly make a wrong decision. To account for the uncertainty in the evidence, we extended the approach by using the Jeffrey's rule (Jeffrey, 1990) to estimate the probability of the best action. Based on this, the probability of a state node X (e.g., $RightArea$ and $LightLocation$), given an action a, is computed by equation 6:

$$P'(X) = \sum_{i=1}^{n} P(X \mid Activity_i, a).P(Activity_i) \tag{6}$$

From equations 1 and 6, EU can then be estimated by equation 7. Note that $Activity$ is no more included in the set of contextual evidence e.

$$EU(a) = \sum_{X} \sum_{i=1}^{n} P(X \mid Activity_i, a, e).P(Activity_i).U(X) \tag{7}$$

6 Experiments

The method was experimented in realistic situations in a smart home with naive users interacting with the environment. This section describes the experimental set up and the results of the decision making.

6.1 Experimental Environment

The pervasive environment considered was the DOMUS Smart Home designed by the *Laboratoire d'Informatique de Grenoble* (LIG) (Gallissot et al., 2013). This flat was extensively used in the SWEET-HOME project for experiments. Figure 8 shows the details of the flat. It is a thirty square meters flat including a bathroom, a kitchen, a bedroom and a study room, all equipped with sensors and actuators such as infra-red movement detectors, contact sensors, video cameras (used only for annotation purpose), etc. In addition, seven microphones were set in the ceiling for audio analysis. The flat is fully usable and can accommodate a dweller for several days. The technical architecture of DOMUS was based on the KNX bus system (`www.knx.org`), a worldwide ISO standard (ISO/IEC 14543) for home and building control. Besides KNX, several field buses coexist in DOMUS, such as UPnP (Universal Plug and Play) for the audio video distribution or X2D for the detection of the opening (doors, windows, and cupboards). More than 150 sensors, actuators and information providers are managed in the flat (e.g., lighting, shutters, security

systems, energy management, heating, etc.). Sounds were recorded independently thanks to a National Instrument mutichannel acquisition board and analysed using PATSH.

The SWEET-HOME system was run on 2 PCs ; one running the PATSH system (audio analysis) and another one to run the intelligent controller. *e-lio*, the communication device used to initiate a communication between a senior and a relative was placed in the study.

The DOMUS flat was designed to seem as normal as a standard flat so that the participants moving in the smart home would behave as naturally as possible.

6.2 Experimental set up and collected data

Two experiments have been carried out in the pervasive environment with people performing activities of daily living. In both cases, the experimenters organised a detailed visit of the flat for each participant to ensure that the flat was familiar to the subject. The main objective of the first experiment was to learn and test the models to infer context, i.e. location, activity, and agitation. In the second experiment, the models for context awareness were used by the intelligent controller and the system was tested in decision making scenarios. This section details the procedures followed for both experiments. In all cases, cameras were used for annotation purposes only.

6.2.1 First Experiment — The Multimodal SWEET-HOME dataset

In this experiment 21 persons (including 7 women) participated to record sensor data in a daily living context. The average age of the participants was 38.5years (min 22, max 63), height 1.72m (min 1.43, max 2.10), and weight 70kg (min 50, max 105). Each individual was asked to perform each of the following activities: (1) to clean, (2) to undress/dress, (3) to prepare a meal and to eat, (4) to perform personal hygiene, (5) to have a phone conversation, (6) to perform leisure activities, (7) to have a nap. In total, for all the participants, more than 26 hours of data have been acquired. During this experiment no vocal orders were used. Participants used switches in order to turn on/off the lights, open the blinds and curtains, and to turn on/off electrical devices. Figure 9 shows the inside of the DOMUS smart home during the experiments.

Every sources of information from the home automation network as well as the audio channels were recorded. This dataset is called the multimodal SWEET-HOME dataset. It was annotated (location, activity, interaction, speech and non speech sounds) using the Advene software[3]. This corpus is freely available at `http://sweet-home-data.imag.fr/`.

Since in this experiment there was no uttered vocal orders, they were artificially included at moments in which a specific situation was encountered. For instance, at the beginning of an eating activity a fake utterance was inserted so as if the user had uttered the vocal order to turn on the light at that moment. Though this is not real utterances, this made it possible to evaluate the situation at each second in the interval of the first 30 seconds of the activity. The situations that were included were the following:

1. **Situation 1**. The user is sitting eating in the kitchen, the most adequate light is the light

[3] `liris.cnrs.fr/advene`

above the table.

2. **Situation 2**. The user is washing up the dishes in the kitchen, the most adequate light is the light above the sink.
3. **Situation 3**. The user is finishing her nap in the bedroom, the most adequate light is the bedside lamp.
4. **Situation 4**. The user is tidying up the kitchen, the most adequate light is the one above the sink.
5. **Situation 5**. The user is reading in the bedroom, the most adequate light is the ceiling light.

6.2.2 Second Experiment — The Interaction SWEET-HOME dataset — Data acquisition for decision making

This experiment has been performed in similar conditions as for the first one. However, in this case, participants were asked to utter vocal orders to activate the actuators in the smart home. The objective of this experiment was to test the intelligent controller *on live*, including the context recognition model, in real situations responding to the vocal orders given by the user. 15 persons (including 9 women) participated to the experiment to record all sensors data in a daily living context. The average age of the participants was 38years (19-62, min-max). At the end of the study, 11 hours of data was recorded. The following activities were performed by the participants: (1) to tidy up the home, (2) to prepare a meal and to eat, (3) to have a conversation by videoconference, (4) to perform leisure activities, (5) to have a nap.

As for the first experiment, the participant were provided with a scenario (divided in 5 parts) to follow . The scenario was designed to last around 45 minutes but there was no constraint on the execution time. Four of the five parts of the scenario were designed to make the user perform daily activities while uttering voice commands. Figure 10 shows the details of the first part of the scenario. The participant is provided with a list of actions to perform and voice commands to utter. As it can be seen, the voice commands were provided but not the grammar. Indeed, the grammar would have been too difficult to manipulate for people unfamiliar with this format.

The remaining part of the scenario was devoted to the command of the *e-lio* system and to the simulation of emergency calls. These scenarios allowed us to process realistic and representative audio events in conditions which are directly linked to usual daily living activities.

Each participant had to use vocal orders to switch the light on or off, open or close blinds, ask about the temperature and ask to call his or her relative. The instruction was given to the participants to repeat the order up to 3 times in case of failure. A wizard of Oz was used in case of persistent problems.

6.3 Situations for Decision Making

Four situations in which the intelligent controller must perform context aware decision making were considered. Each time the vocal order consisted in uttering *"turn on the light"*. In each situation, there are two lights that can be turned on, one brighter than the other:

1. **Situation 1**. The user is sitting eating in the kitchen, the most adequate light is the light above the table.

2. **Situation 2**. The user is tidying up the bedroom, the most adequate light is the ceiling light.

3. **Situation 3**. The user is washing up the dishes in the kitchen, the most adequate light is the light above the sink.

4. **Situation 4**. The user is finishing her nap in the bedroom, the most adequate light is the bedside lamp.

These situations occur when the participant utters a vocal order in an ambiguous situation in which the controller must take into account the contextual information.

6.4 Results

6.4.1 Model learning

Despite the time devoted to the experiment, the dataset was insufficient to learn the weights for an MLN decision model. The building of a real corpus exclusively composed of decision for supervised learning would be a tedious and time consuming task. Thus, the training corpus for weight learning was the result of the simulation of 300 instances, most of which expressing the best location and intensity but also including configurations that were less appropriate. For instance, if in most of the situations where the user is eating in the kitchen, the best light is the one above the table, a light above the sink can also be acceptable. The topology of the flat and the location of the actuators within it helps to find the best configuration of preferences.

6.4.2 Localisation

One of the most important elements of the context is the localisation (i.e., the ability for the system to know in which room the person is). In the case of the Multimodal SWEET-HOME dataset, the performance was of 95% of accuracy. When the localisation was run on the Interaction dataset, the performance was of 85.36% (SD=3.85%) accuracy on average per subject computed every second in each record. This means that nearly 15% of the localisation was wrong. However, when the localisation performance is computed only at the time at which a voice command is recognised (i.e., the location used by the decision module) the performance reaches 100% in both datasets. This means that the system was always making a decision using the correct location information.

6.4.3 Activity Recognition

As discussed before, activity recognition is a difficult task which delivers uncertain information. In this paper, we focus on the activity recognition during a voice command.

Table 2: Confusion matrix of the activity recognition during decision making for the multimodal SWEET-HOME dataset (simulated vocal orders)

Target/Hit	Eat	Tidy	Sleep	Rest	Dress	Hygiene
Eat	492	138	0	0	0	0
Tidy	193	1027	18	5	1	16
Sleep	0	16	611	30	3	0
Rest	0	0	85	415	31	9

The method used to evaluate the MLN activity classifier was based on Cross-Validation but used a specific type namely Leave-One-Subject-Out-Cross-Validation (LOSOCV). If the dataset is composed of records [4] from N participants, for each fold, records from $N-1$ participants were used to train the model, while the remaining record was used for evaluating the learned model. Consequently, testing is done on different individuals from training, and thus LOSOCV prevents participant over-fitting.

The performance of activity recognition with the multimodal SWEET-HOME dataset (cf. Section 6.2.1) is given Table 2. Recall that simulated voice order were placed within a 30-second interval at the beginning of any of the 5 situations for each participant. Thus, activity recognition was evaluated for every second in this 30-second interval. With 21 participants, the total number of times that activity recognition should have been performed would be 3150. However, Situation 5 was not present in a couple of participants' record, therefore the number of instances to evaluate activity recognition was 3090. Table 2 shows the confusion matrix of activity recognition for the multimodal SWEET-HOME dataset (simulated vocal orders). The obtained overall accuracy is 83%.

The audio events are an important source of information to infer the location of the inhabitant, since in the first corpus the vocal order did not take place in reality, there were some temporal windows in which the inferred location was wrong. The location is the most important evidence to infer the activity since given a certain location, the model will give higher probability to activities that are normally performed in that location. It explains why tidy and rest activities have been classified in 25 windows as Hygiene which was always performed in the bathroom which is a room next to the bedroom. The confusion of activities happening in the same room can also be high, this is the case of eating and tidying in the kitchen. In both activities the same contact doors are activated (fridge, shelf), the sounds produced by dishes and cutlery are quite similar, there is just a slight difference in the level of agitation that can help the classifier to discriminate these two activities.

For the second experiment, the interaction SWEET-HOME dataset (cf. Section 6.2.2), that contains genuine vocal orders the confusion matrix is presented Table 3. There were 60 activity instances performed during voice commands, they belong to: sleeping, eating and tidying up. However, our model has been trained to recognize seven activities (the same as (Chahuara et al., 2011)). The most important confusion was between eating and tidying up. Both activity were performed in the kitchen and shared many characteristics such as the noise produced by the dish handling. The voice commands helped to improve the inference of location, that is why in the second experiment there are no confusions with the hygiene activity that is performed in the bathroom. However, there is a confusion in two windows between sleeping and eating, which is produced by a presence of the person in the kitchen just before going to sleep that is considered in the

[4] Here 'record' means, the full data for one human participant

Table 3: Confusion matrix of the activity recognition during decision making – with vocal orders

Target/Hit	Eat	Tidy	Sleep	Rest
Eat	12	3	0	0
Tidy	6	20	3	1
Sleep	2	0	13	0

Table 4: Correct decision rate with and without activity uncertainty – with the multimodal SWEET-HOME dataset

Situation /Expected Utility	MLN without uncertainty	MLN with uncertainty
Situation 1	78%	71%
Situation 2	86%	94%
Situation 3	79%	86%
Situation 4	92%	89%
Situation 5	84%	91%
Total	84%	86%

temporal window. Even more, eating and sleeping have most of the time a low level of agitation what also explain their confusion. The overall accuracy was of 75%.

The overall accuracy of the activity recognition was of 83% and 75%. This is not perfect but it is a reasonable rate given the poverty of the information sources. This also shows the necessity of taking the activity uncertainty into account in the decision model.

6.4.4 Decision Making

Recall from Section 4.3 that the MLN decision model, apart from the localisation and activity, includes a third contextual information: the agitation. This variable domain has then been discretised in three categories: low, medium, and high.

Results obtained with the multimodal SWEET-HOME corpus (simulated vocal orders) are presented Table 4. For each participant and each situation, a voice command was added to the dataset leading to 105 decisions to be made. The second column shows the standard EU, for which the most probable activity is considered as true and others as false. In the third column, the EU is computed using equation 7. In practice, the accuracy of the location was close to 100%, thus the uncertainty was mainly due to the activity recognition. It can be seen that the misclassification between eating and tiding up decreases the performance of the decision making model. However, in most of the situations, considering the uncertainty of the activity recognition output improves the decision taken by the intelligent controller.

Table 5 show the overall correct decision rate for each situation of the corpus with vocal orders. The worst performance is for situations 2 and 3. This is mainly due to the confusion between eating and tidying up. However, the eating activity was well recognized and this explain the high accuracy for situation 1. Overall, the results with and without uncertainty are close. They actually differ in only 3 instances out of 60, 2 of them are produced when the person is eating in the kitchen (situation 1). For instance, in Situation 2 for the participant 12, the activity recognition output was : hygiene(0.20), dressing (0.16), sleeping (0.28), and resting (0.17) while the ground

Table 5: Correct decision rate with and without activity uncertainty – with the interaction SWEET-HOME dataset

Situation /Expected Utility	MLN without uncertainty	MLN with uncertainty
Situation 1	86.6%	100%
Situation 2	73.3%	66.6%
Situation 3	73.3%	73.3%
Situation 4	100%	100%
Total	83.3%	85%

truth was tidying up (0.08) in the kitchen [5]. In this example, there is a high uncertainty about the actual activity, but the most probable activities leading to a high intensity choice for the light, the controller did choose a high intensity despite the most probable activity was sleeping.

7 Discussion and perspective

Decision making in smart homes involves dealing with uncertain, imprecise, and incomplete data while at the same time must allow to be checked and adapted by/for human; and so far, not a single method can overcome all these problems. Therefore, Ambient Intelligence projects must rely on the integration of several methods sharing a common base and serving each one a specific purpose (here location-module, activity-module etc.). The presented framework is a stepping stone towards achieving this goal.

The experimental outcomes shed light on several interesting aspects of the approach for the field of decision making and expert systems. The statistic-relational model used in this work achieves a correct decision rate of 85% overall (71% min, 100% max) showing that it is a relevant model for context-aware decision making from sensors data. The experiment also showed that decision making models can be learned from data as exemplified by the learning of uncertainty weights from a training set. These weights are interpretable by humans. Regarding the modelling of uncertainty in the data (instead of knowledge) the experiment do not exhibit a definite difference against the condition in which uncertainty in data is not modelled. There is however some cases where this modelling does provide improvements which calls for further research in the handling of data and knowledge uncertainty in decision models. Another issue in the current approach is the necessity to discretise the continuous values in the logic rules to make the approach tractable. Although it is not problematic in our application (since discretisation is often performed to improve classification performance (Frank and Witten, 1999)) it might be inadequate for control from continues values and time constrained decision making. Another outcome of the experiment is the evaluation methodology which was long and costly to set up but which has permitted to perform realistic tests of the system. In order to favour reproducibility and comparison of the research, the collected corpus has been made available to the community. This is the only available multi-source sensors dataset with voice order we are aware of (Vacher et al., 2014).

The contributions brought by this study are mainly: (1) the description of a framework for pervasive environments that can be extended to integrate other inference modules than those

[5] It must be emphasized that the activity recognition is performed using a sliding window. In this window several instances of activity can intersect, that is why a sleeping and a hygiene activity can both have a high probability

presented in our work, (2) a decision making method that allows the inclusion of expert knowledge in a declarative fashion while providing very good performance in an uncertain environment as it has been shown in the experimental section, (3) the experimental evaluation of our framework in a real environment with people interacting with the system by vocal orders, which is a major distinction with respect to most works on decision making systems for pervasive environments.

Among the numerous approaches that can be applied to represent knowledge in a pervasive environment, ontologies were used because they comply with the representation of properties we delineated as essential requirements for our framework: readability, modularity and expressibility. The architecture based on this ontological representation is general enough to be adapted and implemented in different pervasive environments varying in room distribution, sensors, and environmental conditions. Besides the flexibility for adaptation, different inference modules among those presented in this work can be integrated while handling their own inference methods or using the ontology reasoner capabilities as the situation recognition module does in this work.

One of the main objectives of decision making in a smart-home is to take the most appropriate action with respect to the situation and the user's goal. In that sense, decision making must be aware of the context in which a decision must be made. In this, work we model the problem through an influence diagram which makes it possible to maximize the utility of an action given the data at hand at the moment of the decision making. The adaptation to the situation is performed by explicit modelling of the situations of interest that are defined within the ontology. However, in realistic setting, the building of such model is confronted with several kinds of uncertainty, two of which being: uncertainty in the knowledge and uncertainty in the input data.

In this work, uncertainty in the decision knowledge is handled using of Markov Logic Networks. According to the performed experiments, MLN seems very promising since it exhibits a high correct decision rate. This model has several advantages for the field of expert systems. Firstly, since MNL is relying on a set of logic formulas it is well suited to intelligent systems where knowledge is represented by means of logic. This can be exploited by translating some of this knowledge from one representation to another to, for instance, adds a priori relational knowledge in the MLN structure. From this perspective, a formal domain knowledge description and logic-based decision method could lead to higher re-usability of the model in another application. Secondly, since MLN is also probabilistic model, it can make inference using uncertain knowledge and with incomplete input. However, as most of probabilistic models, MLN learning requires a large amount of data to estimate the parameters. Unfortunately, corpora with annotated data in smart home environment useful for decision making are sparse. Furthermore, to the best of our knowledge there is no available corpora for decision making from vocal orders. To overcome this limitation, we took benefit from the capacity of the MLN to handle a priori knowledge. It had been possible to acquire the structure from expert knowledge and to estimate the weights from a set of synthetic data. This has proven to be adequate in view of the observed performances of the system. Though not ideal, given the difficulty and cost of acquiring training data in the smart home domain, this way seems promising to alleviate the need of large volumes of training data of purely statistical methods.

As previously stressed, in the smart home environment information is uncertain. Thus, this can have a great impact on the decision process if input data are seen as certain evidence. Since, wrong actions must be avoided, the handling of uncertain evidence must be generalised. The proposed approach is a first step towards this generalisation. Although the decision results with and without the consideration of the activity uncertainty are very close, it had some really positive

impact in some situations.

To further ameliorate our framework, we intend to work on two improvements: the first one relates to a tighter integration of the decision model with the ontology. it would indeed be interesting to address the problem of knowledge maintenance by using an ontology reasoner to check for coherence of the decision model rules. Such an integration would not be trivial since MLN rules are first-order logic rule while description logic and safe rules are only a subset of first-order logic. A second improvement would be to generalise the handling of uncertainty to uncertain evidence.

Acknowledgments

The authors would like to thank the participants who accepted to perform the experiments. Thanks are extended to N. Bonnefond and S. Humblot for their support. This work is supported by the Agence Nationale de la Recherche, under grant ANR-09-VERS-011.

References

Artikis, A., Skarlatidis, A., Portet, F., and Paliouras, G. (2012). Logic-Based Event Recognition. *Knowledge Engineering Review*, 27(4):469–506.

Attard, J., Scerri, S., Rivera, I., and Handschuh, S. (2013). Ontology-based situation recognition for context-aware systems. In *Proceedings of the 9th International Conference on Semantic Systems*, I-SEMANTICS '13, page 113–120, New York, NY, USA. ACM.

Baader, F., Horrocks, I., and Sattler, U. (2008). Description Logics. In van Harmelen, F., Lifschitz, V., and Porter, B., editors, *Handbook of Knowledge Representation*, chapter 3, pages 135–180. Elsevier.

Badii, A. and Boudy, J. (2009). CompanionAble - integrated cognitive assistive & domotic companion robotic systems for ability & security. In *1er Congrés of the Société Française des Technologies pour l'Autonomie et de Gérontechnologie (SFTAG'09)*, pages 18–20, Troyes.

Blachon, D., Cokun, D., and Portet, F. (2014). On-line Context Aware Physical Activity Recognition from the Accelerometer and Audio Sensors of Smartphones. In *European Conference on Ambient Intelligence*, volume 8850 of *Ambient Intelligence*, pages 205–220, Eindhoven, Netherlands. Springer International Publishing.

Carolis, B. D. and Cozzolongo, G. (2004). C@sa: Intelligent home control and simulation. In *International Conference on Computational Intelligence*, pages 462–465.

Chahuara, P., Fleury, A., Portet, F., and Vacher, M. (2016). On-line Human Activity Recognition from Audio and Home Automation Sensors: comparison of sequential and non-sequential models in realistic Smart Homes. *Journal of ambient intelligence and smart environments*, 8(4):399–422.

Chahuara, P., Portet, F., and Vacher, M. (2011). Fusion of Audio and Temporal Multimodal Data by Spreading Activation for Dweller Localisation in a Smart Home. In *STAMI Series, Space, Time and Ambient Intelligence*, Barcelona, Spain.

Chahuara, P., Portet, F., and Vacher, M. (2013). Making Context Aware Decision from Uncertain Information in a Smart Home: A Markov Logic Network Approach. In *Ambient Intelligence*, volume 8309 of *Lecture Notes in Computer Science*, pages 78–93, Dublin, Ireland. Springer.

Charalampos, D. and Maglogiannis, I. (2008). Enabling human status awareness in assistive environments based on advanced sound and motion data classification. In *Proceedings of the 1st international conference on PErvasive Technologies Related to Assistive Environments*, pages 1–8.

Chen, L., Nugent, C. D., Mulvenna, M., Finlay, D., Hong, X., and Poland, M. (2008). A logical framework for behaviour reasoning and assistance in a smart home. *International Journal of Assistive Robotics and Mechatronics*, 9(4):20–34.

Christensen, H., Casanueva, I., Cunningham, S., Green, P., and Hain, T. (2013). homeservice: Voice-enabled assistive technology in the home using cloud-based automatic speech recognition. In *4th Workshop on Speech and Language Processing for Assistive Technologies*, pages 29–34.

Cristoforetti, L., Ravanelli, M., Omologo, M., Sosi, A., Abad, A., Hagmueller, M., and Maragos, P. (2014). The DIRHA simulated corpus. In *The 9th edition of the Language Resources and Evaluation Conference (LREC)*, pages 2629–2634, Reykjavik, Iceland.

Dey, A. K. (2001). Understanding and using context. *Personal Ubiquitous Computing*, 5(1):4–7.

Filho, G. and Moir, T. J. (2010). From science fiction to science fact: a smart-house interface using speech technology and a photorealistic avatar. *International Journal of Computer Applications in Technology*, 39(8):32–39.

Frank, E. and Witten, I. H. (1999). Making better use of global discretization. In *Proc 16th International Conference on Machine Learning*, Bled, Slovenia, pages 115–123. Morgan Kaufmann.

Gallissot, M., Caelen, J., Jambon, F., and Meillon, B. (2013). Une plateforme usage pour l'intégration de l'informatique ambiante dans l'habitat. l'appartement domus. *Technique et Science Informatiques (TSI)*, 32(5):547–574.

Gemmeke, J. F., Ons, B., Tessema, N., Van Hamme, H., Van De Loo, J., De Pauw, G., Daelemans, W., Huyghe, J., Derboven, J., Vuegen, L., Van Den Broeck, B., Karsmakers, P., and Vanrumste, B. (2013). Self-taught assistive vocal interfaces: an overview of the aladin project. In *Interspeech 2013*, pages 2039–2043.

Gómez-Romero, J., Serrano, M. A., Patricio, M. A., García, J., and Molina, J. M. (2012). Context-based scene recognition from visual data in smart homes: an information fusion approach. *Personal and Ubiquitous Computing*, 16(7):835–857.

Hamill, M., Young, V., Boger, J., and Mihailidis, A. (2009). Development of an automated speech recognition interface for personal emergency response systems. *Journal of NeuroEngineering and Rehabilitation*, 6.

Helaoui, R., Niepert, M., and Stuckenschmidt, H. (2011). Recognizing interleaved and concurrent activities using qualitative and quantitative temporal relationships. *Pervasive Mob. Comput.*, 7(6):660–670.

Helaoui, R., Riboni, D., and Stuckenschmidt, H. (2013). A probabilistic ontological framework for the recognition of multilevel human activities. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 345–354, New York, NY, USA. ACM.

Hitzler, P. and Parsia, B. (2009). Ontologies and rules. In Staab, S. and Studer, R., editors, *Handbook on Ontologies*, pages 111–132. Springer, Heidelberg, 2nd edition.

Howard, R. and Matheson, J. (1981). Influence diagrams. *Readings on The Principles and Applications of Decision Analysis*, 1 and 2:720.

Istrate, D., Vacher, M., and Serignat, J.-F. (2008). Embedded implementation of distress situation identification through sound analysis. *The Journal on Information Technology in Healthcare*, 6:204–211.

Jeffrey, R. C. (1990). *The Logic of Decision*. University Of Chicago Press.

Katzouris, N., Artikis, A., and Paliouras, G. (2014). Event recognition for unobtrusive assisted living. In *8th Hellenic Conference on AI*, pages 475–488, Ioannina, Greece. Springer-Verlag.

Klein, M., Schmidt, A., and Lauer, R. (2007). Ontology-centred design of an ambient middleware for assisted living: The case of soprano. In *30th Annual German Conference on Artificial Intelligence (KI 2007)*.

Kofler, M. J., Reinisch, C., and Kastner, W. (2012). A semantic representation of energy-related information in future smart homes. *Energy and Buildings*, 47:169–179.

Lacey, G. and MacNamara, S. (2000). Context-aware shared control of a robot mobility aid for the elderly blind. *I. J. Robotic Res.*, 19(11):1054–1065.

Lecouteux, B., Vacher, M., and Portet, F. (2011). Distant Speech Recognition in a Smart Home: Comparison of Several Multisource ASRs in Realistic Conditions. In *Interspeech 2011*, pages 2273–2276, Florence, Italy.

Lee, S.-H. and Cho, S.-B. (2012). Fusion of modular bayesian networks for context-aware decision making. In Corchado, E., Snášel, V., Abraham, A., Wozniak, M., Graña, M., and Cho, S.-B., editors, *Hybrid Artificial Intelligent Systems*, volume 7208 of *Lecture Notes in Computer Science*, pages 375–384. Springer Berlin / Heidelberg.

Leong, C. Y., Ramli, A., and Perumal, T. (2009). A rule-based framework for heterogeneous subsystems management in smart home environment. *IEEE Transactions on Consumer Electronics*, 55(3):1208–1213.

Liao, H.-C. and Tu, C.-C. (2007). A RDF and OWL-Based Temporal Context Reasoning Model for Smart Home. *Information Technology Journal*, 6(8):1130–1138.

Loke, S. W. (2004). Representing and reasoning with situations for context-aware pervasive computing: A logic programming perspective. *Knowl. Eng. Rev.*, 19(3):213–233.

Lowd, D. and Domingos, P. (2007). Efficient weight learning for markov logic networks. In *In Proceedings of the Eleventh European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 200–211.

Mileo, A., Merico, D., and Bisiani, R. (2010). Support for context-aware monitoring in home healthcare. *J. Ambient Intell. Smart Environ.*, 2(1):49–66.

Mileo, A., Merico, D., and Bisiani, R. (2011). Reasoning support for risk prediction and prevention in independent living. *Theory Pract. Log. Program.*, 11(2-3):361–395.

Mitra, K., Zaslavsky, A. B., and Åhlund, C. (2011). A probabilistic context-aware approach for quality of experience measurement in pervasive systems. In *SAC*, pages 419–424.

Moore, P., Hu, B., and Jackson, M. (2011). Rule strategies for intelligent context-aware systems: The application of conditional relationships in decision-support. In *International Conference on Complex, Intelligent and Software Intensive Systems, CISIS 2011*, pages 9–16, Seoul, Korea.

Nath, A. and Domingos, P. (2009). A language for relational decision theory. In *Proceedings of the Sixth International Workshop on Statistical Relational Learning*, Leuven, Belgium.

Nath, A. and Domingos, P. (2010). Efficient belief propagation for utility maximization and repeated inference. In *AAAI*.

Nishiyama, T., Hibiya, S., and Sawaragi, T. (2011). Development of agent system based on decision model for creating an ambient space. *AI & Society*, 26(3):247–259.

OWL2 (2012). Owl 2 web ontology language document overview (second edition). Technical Report http://www.w3.org/TR/2012/REC-owl2-overview-20121211/, W3C.

Pan, R., Peng, Y., and Ding, Z. (2006). Belief update in bayesian networks using uncertain evidence. In *18th IEEE International Conference on Tools with Artificial Intelligence*, pages 441–444.

Popescu, M., Li, Y., Skubic, M., and Rantz, M. (2008). An acoustic fall detector system that uses sound height information to reduce the false alarm rate. In *Proc. 30th Annual Int. Conference of the IEEE-EMBS 2008*, pages 4628–4631.

Portet, F., Vacher, M., Golanski, C., Roux, C., and Meillon, B. (2013). Design and evaluation of a smart home voice interface for the elderly: acceptability and objection aspects. *Personal and Ubiquitous Computing*, 17:127–144.

Ranganathan, A. and Campbell, R. H. (2003). An infrastructure for context-awareness based on first order logic. *Personal Ubiquitous Computing*, 7(6):353–364.

Richardson, M. and Domingos, P. (2006). Markov logic networks. *Machine Learning*, 62(1-2):107–136.

Rodríguez, N. D., Cuéllar, M. P., Lilius, J., and Calvo-Flores, M. D. (2014). A fuzzy ontology for semantic modelling and recognition of human behaviour. *Knowl.-Based Syst.*, 66:46–60.

Rodríguez, N. D., Cuéllar, M. P., Lilius, J., and Calvo-Flores, M. D. (2014). A survey on ontologies for human behavior recognition. *ACM Computing Surveys*, 46(4):43:1–43:33.

Schilit, B., Adams, N., and Want, R. (1994). Context-aware computing applications. In *In Proceedings of the Workshop on Mobile Computing Systems and Applications*, pages 85–90. IEEE Computer Society.

Sehili, M. E. A., Lecouteux, B., Vacher, M., Portet, F., Istrate, D., Dorizzi, B., and Boudy, J. (2012). Sound environment analysis in smart home. In *AmI '12 : International Joint Conference on Ambient Intelligence*, volume 7683, pages 208–223, Pisa, Italy. Springer.

Vacher, M., Caffiau, S., Portet, F., Meillon, B., Roux, C., Elias, E., Lecouteux, B., and Chahuara, P. (2015). Evaluation of a context-aware voice interface for Ambient Assisted Living: qualitative user study vs. quantitative system evaluation. *ACM - Transactions on Speech and Language Processing*, 7(issue 2):5:1–5:36.

Vacher, M., Lecouteux, B., Chahuara, P., Portet, F., Meillon, B., and Bonnefond, N. (2014). The Sweet-Home speech and multimodal corpus for home automation interaction. In *The 9th edition of the Language Resources and Evaluation Conference (LREC)*, pages 4499–4506, Reykjavik, Iceland.

Vacher, M., Lecouteux, B., Istrate, D., Joubert, T., Portet, F., Sehili, M., and Chahuara, P. (2013). Experimental Evaluation of Speech Recognition Technologies for Voice-based Home Automation Control in a Smart Home. In *4th Workshop on Speech and Language Processing for Assistive Technologies*, pages 99–105, Grenoble, France.

Vacher, M., Portet, F., Fleury, A., and Noury, N. (2011). Development of audio sensing technology for ambient assisted living: Applications and challenges. *International Journal of E-Health and Medical Communications*, 2(1):35–54.

Wolf, P., Schmidt, A., and Klein, M. (2008). Soprano - an extensible, open aal platform for elderly people based on semantical contracts. In *3rd Workshop on Artificial Intelligence Techniques for Ambient Intelligence (AITAmI'08), 18th European Conference on Artificial Intelligence (ECAI 08), Patras, Greece*.

Yau, S. S. and Liu, J. (2006). Hierarchical situation modeling and reasoning for pervasive computing. In *Proceedings of the The Fourth IEEE Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, and the Second International Workshop on Collaborative Computing, Integration, and Assurance (SEUS-WCCIA'06)*, SEUS-WCCIA '06, pages 5–10, Washington, DC, USA. IEEE Computer Society.

Yilmaz, O. and Erdur, R. C. (2012). iConAwa – an intelligent context-aware system. *Expert Systems with Applications*, 39(3):2907–2918.

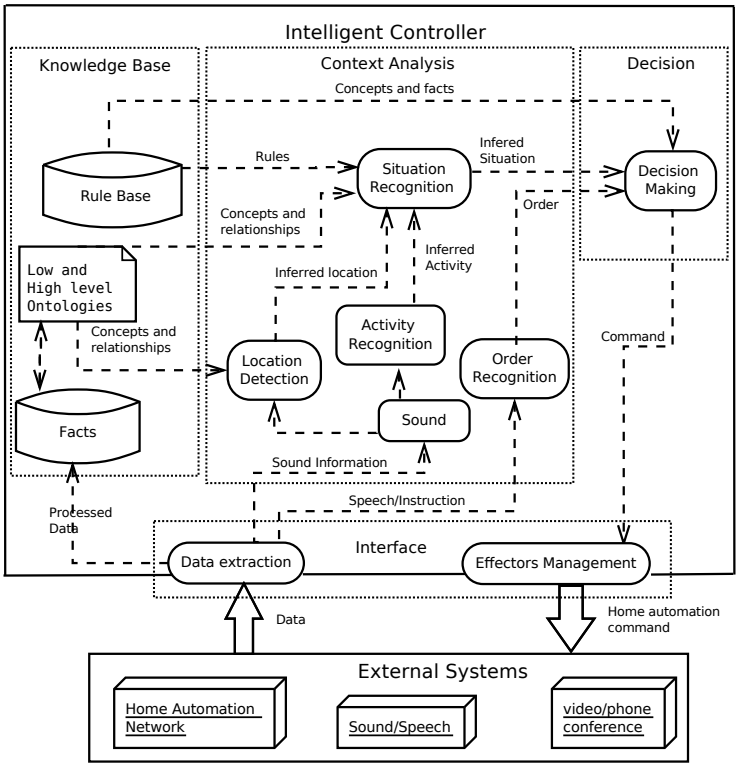


Figure 2: The Intelligent Controller Diagram

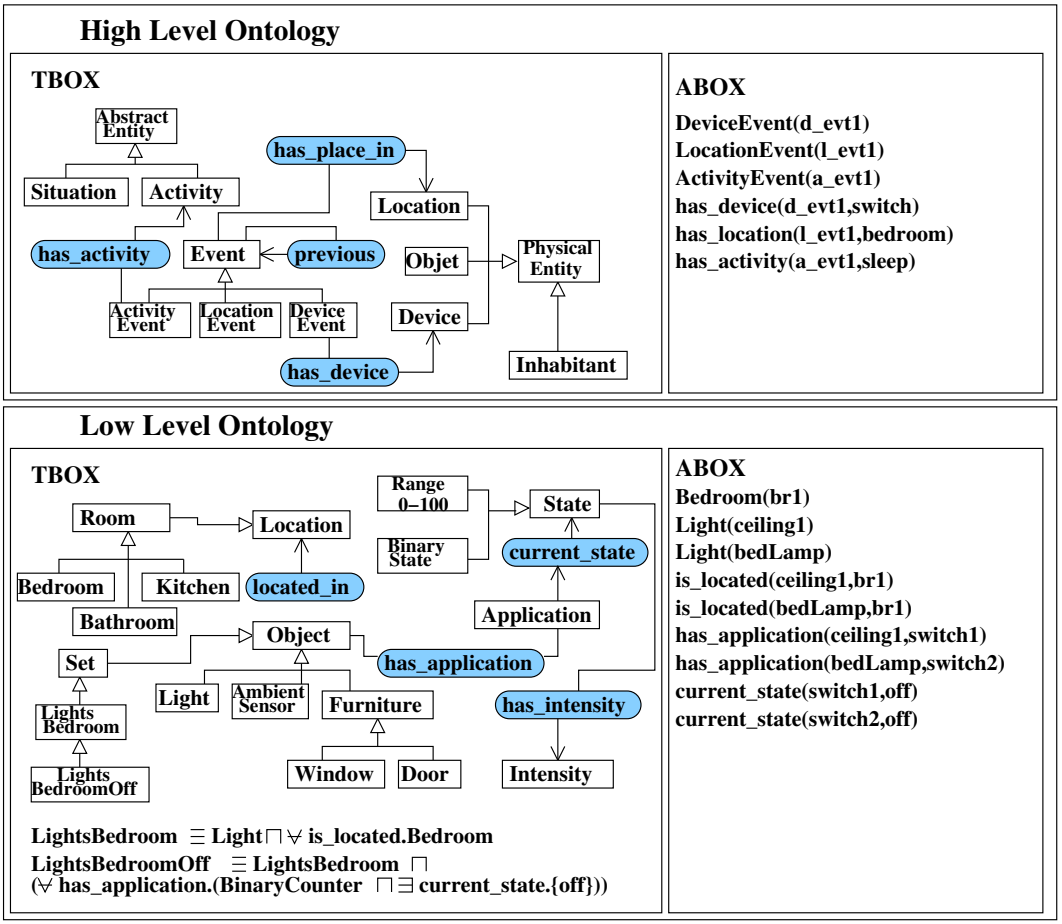


Figure 3: The low and high level ontologies.

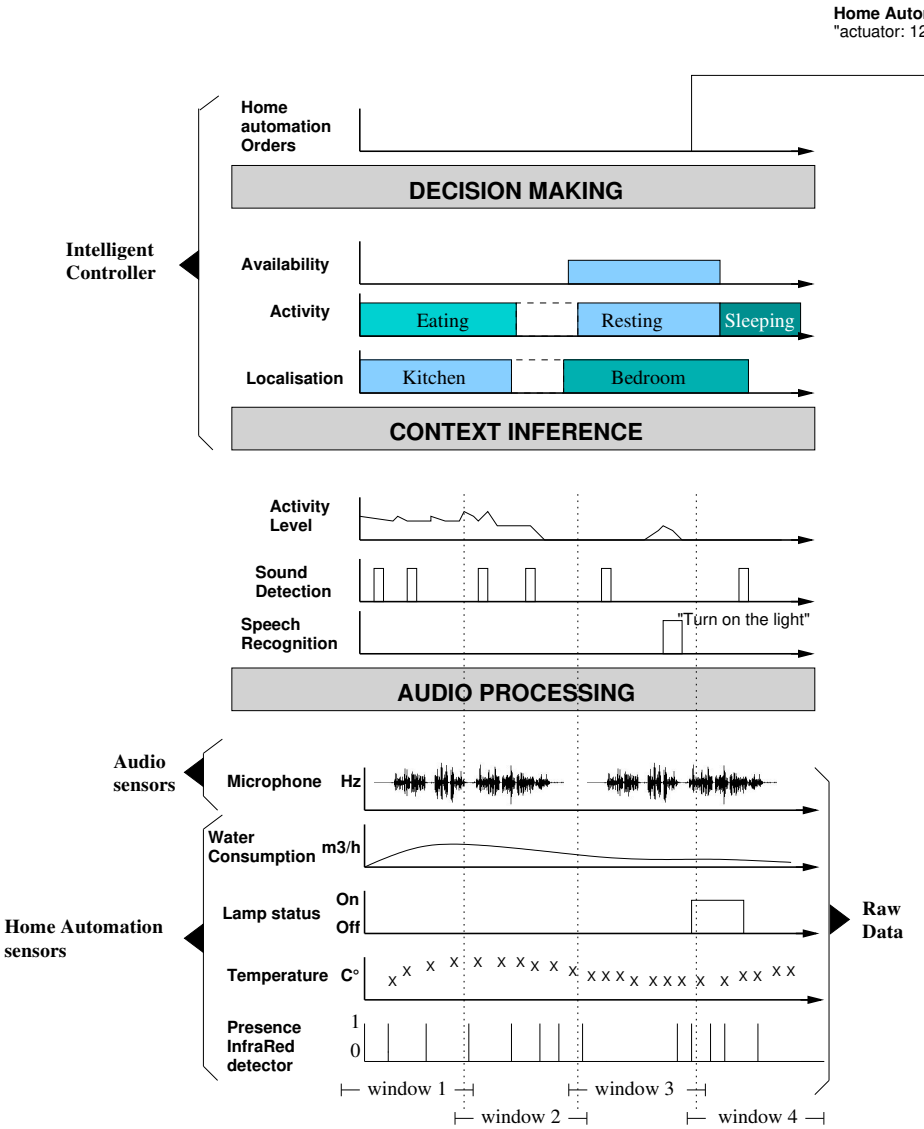


Figure 4: Inference of information from sensor raw data.

```
basicCmd          = key initiateCommand object |
                    key emergencyCommand
key               = "Nestor"
initiateCommand   = "ouvre" | "ferme" | "baisse" | "éteins" | "monte" | "allume" | "descend" |
                    "appelle" " | "donne"
emergencyCommand  = "au secours" | "à l'aide"
object            = [determiner] ( device | person | organisation)
determiner        = "mon" | "ma" | "l'" | "le" | "la" | "les" | "un" | "des" | "du"
device            =  "lumière" | "store" | "rideau" | "télé" | "télévision" |
                     "radio" | "heure" | "température"
person            = "fille" | "fils" | "femme" | "mari" | "infirmière" | "médecin" | "docteur"
organisation      = "samu" | "secours" | "pompiers" | "supérette" | "supermarché"
```

Figure 5: Excerpt of the voice orders grammar (terminal symbols are in French)

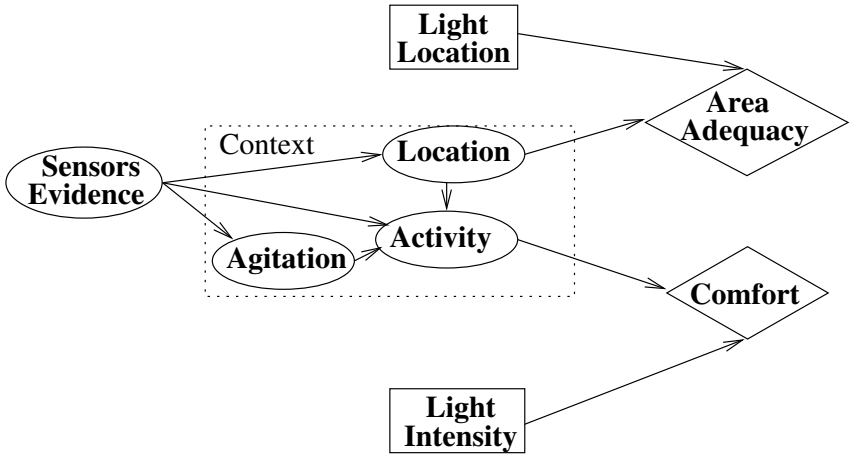


Figure 6: Influence diagram for a decision after a vocal order is recognised.

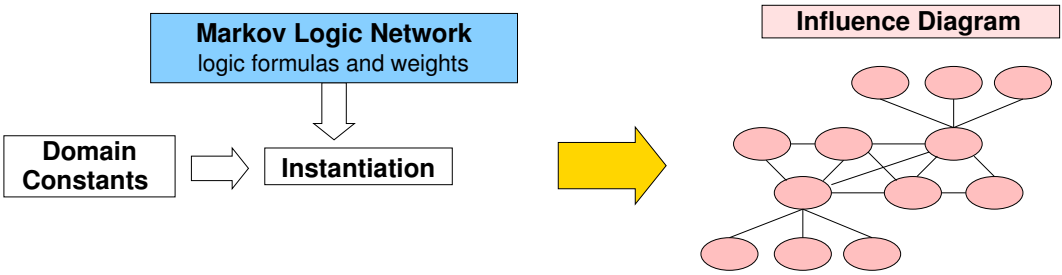


Figure 7: Influence diagram construction by MLN grounding.

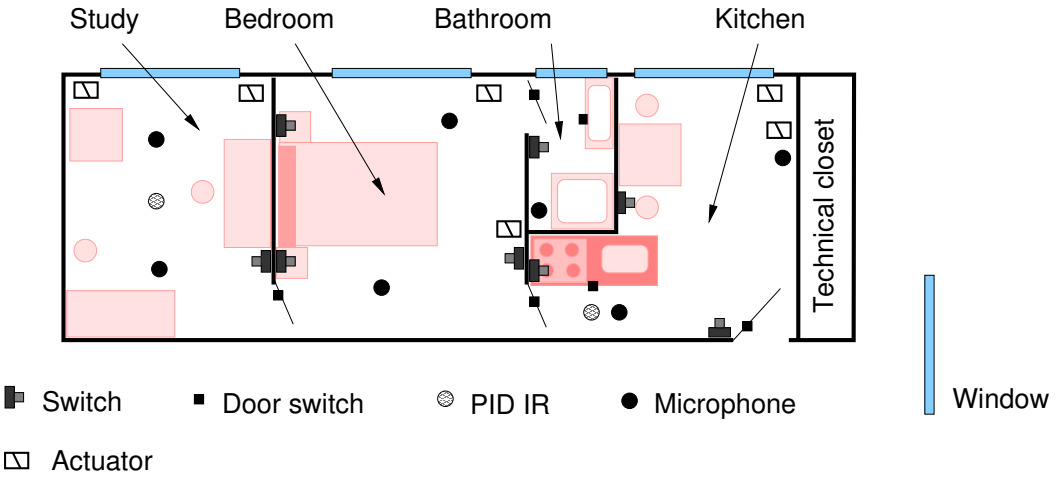


Figure 8: The DOMUS Smart Home of the LIG (Vacher et al., 2014)



Figure 9: Images captured in the smart home

Go to the kitchen
Ask for the ambient temperature:
Nestor donne la température (*Nestor give the temperature*)
You can have a snack
Once finished, put the dishes in the sink
Ask for the current time:
Nestor donne moi l'heure (*Nestor give the time*)
You realise that it is late, you must go shopping
before leaving the home, you want to turn off the light:
Nestor éteins la lumière (*Nestor shut off the light*)
you also want to close the blinds:
Nestor baisse les stores (*Nestor close the blinds*)
Finally you leave the home

Figure 10: Excerpt of the first scenario given to the participant



Figure 11: Screenshots of the recording video cameras.