Context Aware Decision System in a Smart Home: Knowledge Representation and Decision Making Using Uncertain Contextual Information

Pedro Chahuara, François Portet, and Michel Vacher *

Laboratoire d'Informatique de Grenoble,
UMR 5217 CNRS/UJF/G-INP, FRANCE
{pedro.chahuara,francois.portet,michel.vacher}imag.fr

Abstract. This research addresses the issue of building home automation systems reactive to voice for improved comfort and autonomy at home. The paper presents a complete framework that acquires data from sensors and interprets them, by means of IA techniques, to provide contextual information for decision making. The system uses a two-level ontology to represent the different concepts handled during the processing which also contains SWRL instances to automatise some of the reasoning. The focus of this paper is on the relationship between the knowledge representation and the decision process which uses a dedicated Markov Logic Network approach to benefit from the formal logical definition of decision rules as well as the ability to handle uncertain facts inferred from real data. The entire approach is situated w.r.t. the Sweet Home project whose aim is to make possible context-aware voice command at home.

Keywords: Ambient intelligence and pervasive computing, Decision making, Frameworks for formalizing context and context-aware knowledge representation, Reasoning under uncertainty

1 Introduction

As the development of Smart Homes (SH) has gained a growing interest among many communities — such as medicine, architecture, computer sciences, etc. — two major challenges have emerged in the area of Ambient Intelligence. Firstly, the need for knowledge representation models featuring high readability, modularity and expressibility. Secondly, the requirement to develop decision making methods that can leverage knowledge models to take context — the particular situation under which a decision is taken — and its uncertainty into account. Indeed, in most cases the information gathered to infer context comes from sources affected by uncertainty and imprecision.

In the literature, logical models, mostly ontologies and logic rules, seem to have reach a consensus due to the high readability and expressibility they offer.

* This work is part of the Sweet-Home project founded by the French National Research Agency (Agence Nationale de la Recherche / ANR-09—VERS-011)

The Open AAL platform [19] uses an ontology that describes in-home entities belonging to low and high abstraction levels. The framework designed around this ontology is appropriate to facilitate the integration of devices from different providers, as they share a common taxonomy, and the implementation of computational methods to make context inference. The independence between knowledge representation and inference methods guarantees modularity, however it does not take advantage of the reasoning capacities supported by logical reasoners, as the only purpose of the ontology is to be an artefact of integration. Chen et al. [3] have proposed a method to perform activity recognition in home, an important element of context awareness, by using subsumption checking in an ontology, but uncertainty is not supported in this work. A more general approach was designed by Liao [9], in which some context elements, such as level of risk, are defined through logic rules using RDF-based events to perform activity recognition. However, uncertainty of the information sources is not considered even if a prior probability of risk is estimated. Answer Set Programming (ASP) is another logic approach for representation and reasoning that has been applied by Mileo et al. [11] to estimate the evolution of the inhabitant's health state. They present a framework that can properly deal with reasoning under incompleteness and uncertainty. Furthermore, the knowledge encoded in the ASP rules could be integrated into an ontology as well. Although their approach is very relevant for context recognition, they have not developed formal decision models containing essential elements such as utilities, risks and actions. On the side of decision methods for SH dealing with uncertainty, several Bayesian approaches have been suggested, as in the SOCAM project [5]. Influence diagrams [7], which are based on Bayesian networks, have been also applied to model the causal relation among decision actions, uncertain variables, risk, and utilities [14, 4]. However in these works, the decision process is not supported by a formal knowledge representation that can be exploited in other tasks besides decision.

It seems that there exists a gap between the development of formal models to represent knowledge in pervasive environments and the methods for decision making that must act under uncertain information. We are tackling this problem in the Sweet-Home project, a new smart home system whose main man-machine interaction modality is based on audio processing technology. Our proposed solution involves the representation of concepts by means of ontologies and a set of logical rules. It takes advantage of description logic reasoners and SWRL for situation recognition and obtains a system adaptable to other SH implementations, as well. In the decision stage, a part of the logical rules is employed 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. The rest of this paper describes the Sweet-Home framework. Sections 2 presents the project and Section 3 the framework architecture. Section 4 shows the ontologies and how situation recognition is performed. A detailed explanation of our decision making model is given in Section 5. Finally Section 6 concludes with a brief discussion.

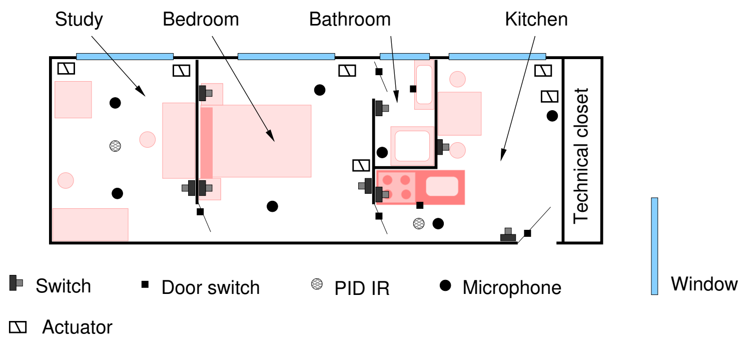

Fig. 1. The DOMUS smart home.

2 The Smart Home context

This research is related to the Sweet-Home project (`http://sweet-home.imag.fr`), a French national supported research project aiming at designing a new smart home system based on audio technology which focuses on three main aspects: to provide assistance via *natural human-machine interaction* (voice and tactile command), to ease *social interaction* and to provide *security reassurance* by detecting situations of distress. In this project, the SH under consideration is DOMUS, a flat filled in with sensor technology which was set up by the Multicom team of the Laboratory of Informatics of Grenoble. This $30m^2$ suite flat, depicted in Figure 1, is equipped with sensors and actuators such as infra-red presence detectors, contact sensors, video cameras (used only for annotation purpose), etc.

This kind of smart home can support daily living by making context-aware decision base on the current situation the user is. To illustrate this support let's consider the following two scenarios:

Scenario 1 *The inhabitant arrives to her apartment at night and goes to the bedroom immediately, forgetting to lock the door. 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, at this moment, from the sequence of her interactions with the environment, the system recognizes that she is about to sleep, and a relatively dangerous situation is recognized as the main door is not locked. A decision could result in sending a message through a speech synthesizer – considering the risk of interrupting her rest– to remind her of the state of the door.*

Scenario 2 *The inhabitant wakes up in the middle of the 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 what the appropriate intensity is. 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.*

From these scenarios it can be noticed that contextual information, such as location and activity, plays a major role to deliver appropriate support to the user. In this paper 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 live; such as, sleeping, cooking, or cleaning. In an instant t the activity might be undetermined; so an activity occurrence, a is defined in an interval of time, $A(t_{begin}, t_{end})$. Thus $A : t_b, t_e \rightarrow a$, $t_b, t_e \in \mathbb{N}$ and $t_b < t_e$*

Moreover, many more information can be inferred from the raw data such as agitation, communication, etc. They are defined as sources of information:

Definition 3 (Source of Information). *The system contains a set of variables V that describes 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 a 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 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 $C_i^{k_i}$ establish a set $A_i \subset DOM(V_{k_i})$ to constraint the value of a source of information V_{k_i}. Thus $S = \{v / \forall C_i^{k_i} \in C, v_{k_i} \in A_i\}$*

For example, if we have two sources of information, V_1 and V_2, corresponding to the the state of the main door and the location of the inhabitant, a situation can be defined by constraints, C_1^1, C_2^2, holding the following sets: $A_1 = \{open\}, A_2 = \{study, bedroom\}$.

Definition 6 (Temporal Situation). *Let's consider a temporal sequence of system states $\delta = (v_1^{t_1}, v_2^{t_2}, ..., v_n^{t_n})$ where t_i is the time of occurrence. A temporal situation R, is defined by a set of constraints $T = \{T_1, T_2, ..., T_m\}$, where each T_k defines a pair of situations (S_k^1, S_k^2) and an interval $[a_k, b_k]$ such as $R = \{(v_i^{t_i}, v_j^{t_j}) / \forall T_k \in T, v_i^{t_i}, v_j^{t_j} \in \delta, v_i \in S_k^1, v_j \in S_k^2, a_k \leqslant t_j - t_i \leqslant b_k\}$*

Thus, if a temporal constraints T_1 establish an interval $[t_i, t_j]$, a temporal situation will be recognized when two instances of the situations S^1 and S^2 occur with a difference of time falling into the interval. In the rest of the paper we refer to temporal situations simply as situations.

Based on our study of the context, we define it as follows:

Definition 7 (Context). *Set of informations characterizing the circumstance under which an inference is made.*

The main usage of context is disambiguation. When a situation is recognized, context provides the complementary information to evaluate the circumstance in terms of a certain quality $Q \in \{risk, comfort, safety, ...\}$. Let's a function F_Q assigning a value, in the scope of Q, to a situation S. The final value of F_Q depends on the information contained in the context $\kappa : F_Q(S|\kappa)$. This function in our work is given by the decision model.

3 The Sweet-Home System: an Audio-controlled Smart Home

The input of the Sweet-Home system is composed of the information from the domotic system transmitted via a local network and information from the microphones transmitted through radio frequency channels. The microphone data is processed by an audio processing chain delivering hypotheses about the sound or the sentences being uttered by the user [18]. All these streams of information (audio and domotic) are captured by an intelligent controller which interprets them to recognize situations and makes decisions. The diagram of this intelligent controller is depicted in Figure 2. The knowledge of the controller is defined using two semantic layers: the *low-level* and the *high-level* ontologies which are described in the next section. Besides knowledge representation, another role of the ontologies is to store the events from which inference is carried out.

The estimation of the current situation is carried out through the collaboration of several processors, each one being specialized in a specific source of information. All processors share the knowledge specified in both 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.

We have considered that the main aspects for situation recognition 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. For example, in Scenario 2, when the vocal order *Turn on the light* is uttered by the inhabitant, in order to decide which light must be activated, the controller infers inhabitant's location. Furthermore, there can be many lights in the same room, so if the command is given in the middle of the night after the inhabitant has interrupted her sleep, knowing the previous activity and time period helps to infer that the best choice of light are the bedside lamps. Other works have also reckoned location and activity as fundamental for context inference [11, 16].

In order to perform location and activity inference, two independent modules were developed and integrated in the framework. Due to space limitation the reader is referred to [2, 1] for further details.

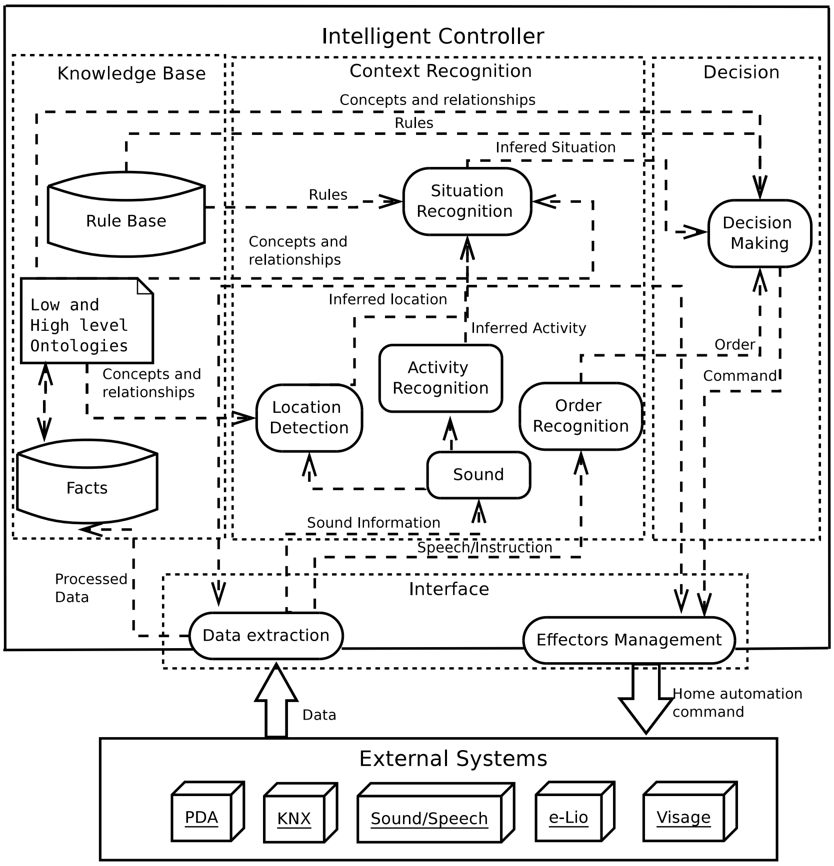

Fig. 2. The Intelligent Controller Diagram.

4 Ontologies and Rules for Situation Recognition

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 a sensor information. These events are considered of low level as they do not require inference. Once they are stored in the facts base, processing modules are executed hierarchically (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 two ontologies were designed, not only for domain knowledge representation, but also for storing the events resulting from the processing modules. Furthermore, situations are defined within the ontologies allowing description logic reasoners to evaluate if a situation is happening. Consequently, the importance of the ontology transcends the mere description of the environment.

4.1 Low and High Level Ontologies

The knowledge of the controller is defined using two semantic layers: the *low-level* and the *high-level* ontologies. The former 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 makes possible a higher re-usability of the reasoning layer when the sensor network and the home must be adapted [8].

Figure 3 shows some of the concepts and relations of both ontologies. The ABoxes serve as an example of the state of the fact base at a certain moment. Let's refer to the scenario 1 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 room is off. In the high level ontology, the interaction with the switch lamp is stored as a device event having time and room as properties. At this stage, the module on charge of location is requested and it gives a straightforward answer as the switch is placed in the bedroom. Then, the evidences 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. 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 will be labelled as detected in the ontology. Detected situations are treated by the decision module explained in section 5.

4.2 Application of SWRL to Situation Recognition

A situation can be seen as a temporal pattern of the system state which is given by the facts base. Ontologies provide an appropriate foundation for situation recognition since they store all the facts and 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 transformed in description logic and written on ontologies [6]. However the scope of this approach is very limited as it does not allow to specify complex definitions. Even when it is limited to safe rules, it overcomes several restrictions of description logics while having the definitions still as part of the ontology. In addition, SWRL builtin functions further extend the semantics of context definitions.

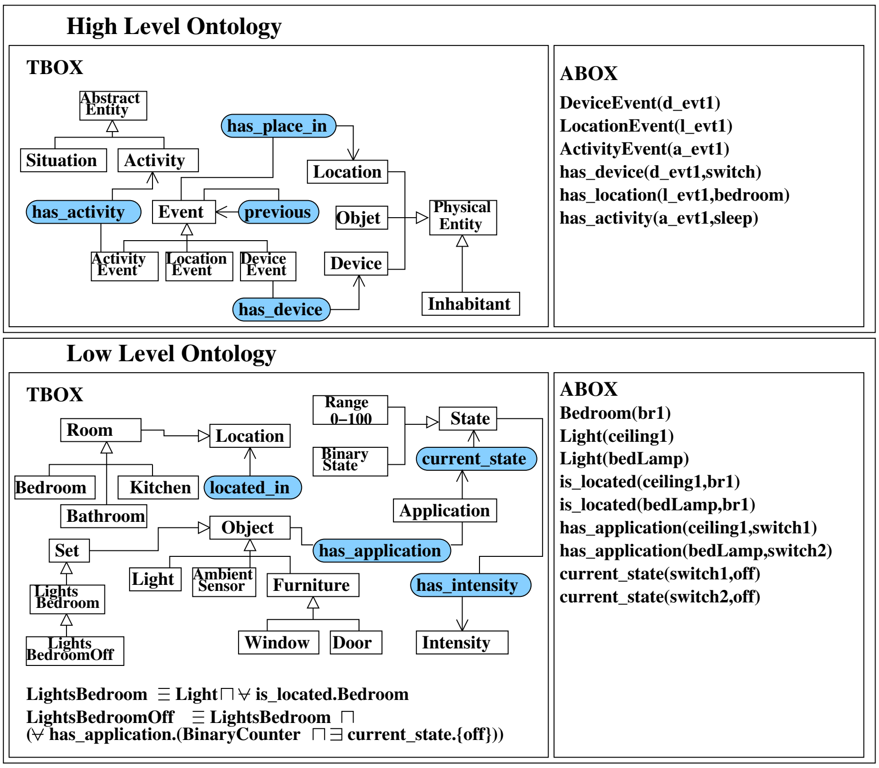

Fig. 3. The low and high level ontologies.

A possible situation definition in SWRL, based on scenario 1, is given below:
DeviceEvent(?d), has_associated_object(?d, door),
takes_place_in(?d, kitchen),state_value(?d, open),
DeviceEvent(?l), has_associated_object(?l, setLights),
takes_place_in(?l, kitchen),state_value(?l, off), temp:after(?l,?d)
$\rightarrow$ *current_state(LightsOffOpenMainDoor, detected)*

We assume that these events reflect the current state of the system. Note that a high level, events can be defined by means of sets of devices as well.

5 Decision Making

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. In this section we briefly describe the base method used for decision making, and give details about our implementation.

5.1 Markov Logic Networks (MLN)

MLN [15] 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 soften 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 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. A MLN, however, can also have hard constraints by giving a infinite weight to some formulas, so that worlds violating these formulas have zero probability. 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. During the inference process [15], every MLN predicated is grounded and Markov network $M_{F,C}$ is constructed where each random variable corresponds to a ground atom. The obtained Markov network allows to estimate the probability of a possible world $P(X = x)$ by the equation 1:

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

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. Exact inference in MLN is intractable in most cases, so Markov Chain Monte Carlo methods are applied [15].

Learning an MLN consists of 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. The most applied algorithm in the literature is *Scaled Conjugate Gradient* [10].

5.2 Influence Diagrams with MLN

Influence diagrams [7] are probabilistic models used to represent decision problems. They result from an extension of Bayesian networks – composed only of state nodes – by the inclusion of two types of node: actions and utilities. An action node is a variable corresponding to a decision choice. The state nodes in the Bayesian network represent how the variables in the problem domain are affected by the actions. Finally, utility nodes are variables that represent the value obtained as consequence of decisions made. Formally, given a set of actions A, an assignment of choices to these actions a, $a \in A$, is taking according to its utility function, $U : X \to [0, 1]$, where X is the state of the random variables in the network after the decision is made. The expected utility for the assignment of choices a is computed as: $EU(a) = \sum_X P(X|a, e)U(X)$ Where e is the evidence given to the network. The process of finding the optimal decision, i.e.

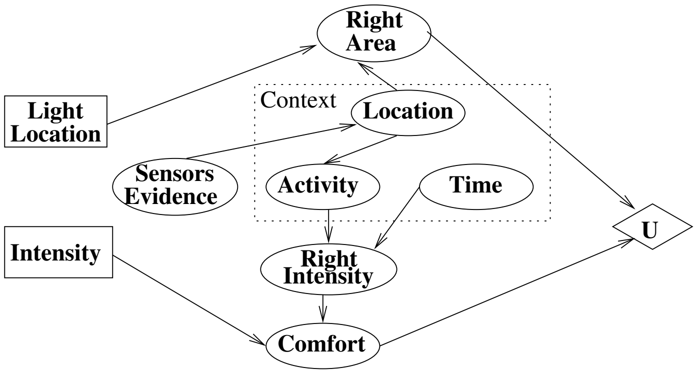

Fig. 4. Influence diagram for a decision after a vocal order is recognised.

the assignment of choices to actions, consists of solving the Maximum Expected Utility (MEU) problem which demands to compute every possible assignment: $argmax_a EU(a)$.

Figure 4 shows an example of Influence Diagram, based on the scenario 2, where a decision is made as a response to a vocal order *Turn on the light*. In this case, the setting of action variables, represented by rectangular nodes, designates which lights devices in the environment are operated and the intensity of the lights. Round nodes are the state nodes affected by the decision. Among the state nodes, information belonging to the context is bound within a dashed area. There are two variables influencing directly the utility: the comfort of the inhabitant and the suitability of the activated lights location that ideally should be the same of the inhabitant. Note that this location is not easy to determine in some cases since the inhabitant could be moving in the environment while uttering the vocal order.

Since a Markov network is a more general probabilistic model than a Bayesian network, Influence diagrams can also be implemented by means of MLN [12]. Nath et al. [13] 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 assignation. We have considered this approach suitable for implementing decision making in our framework for two main reasons: First at all, MLNs are defined through logical rules which can be stored in an ontological representation, using the concepts already established in order to keep a standard vocabulary besides achieving decision model readability. Secondly, it allows to deal with the uncertainty related to context variables and evidence.

A MLN for the influence diagram in figure 4 can be defined as follows:

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

Weight	**Rule**
2.0	$LightLocation(l) \wedge Location(l) \rightarrow RightArea(good)$
1.8	$LightLocation(l1) \wedge Location(l2) \wedge NextTo(l1, l2)$ $\rightarrow RightArea(acceptable)$
2.0	$Intensity(d) \wedge Activity(a) \wedge RightIntensity(a, d)$ $\rightarrow Comfort(high)$
1.2	$Intensity(d1) \wedge Activity(a) \wedge RightIntensity(a, d2) \wedge d1! = d2$ $\rightarrow Comfort(bad)$

Utility Values

U(RightArea(bad))=-1	U(RightArea(acceptable))=0	U(RightArea(good))=1
U(Comfort(low))=-1	U(Comfort(medium))=0	U(Comfort(high))=1

Evidences(When they are not factual, then probability is indicated)

Location(bedroom)[0.8]	Location(kitchen)[0.15]	Location(toilet)[0.05]
Activity(sleep)[0.75]	Activity(read)[0.17]	Activity(dress)[0.08]
RightIntensity(sleep,low)[0.95]	RightIntensity(read,low)[0.80]	
NextTo(kitchen,bedroom)	NextTo(bedroom,toilet)	

This model must be constructed dynamically since the probability of context variables, location and activity, can not be known *a priori*. These variables are provided by the specialised modules of location and activity that supply also a probability for their inference results. These results are uncertain evidences. To introduce them into the MLN model, we have followed an approach similar to the one implemented by Trans et al. [17]. Therefore, after the vocal command is received, the context variable values are requested by the decision module, the decision model is constructed using the rules saved in the ontology and decision inference is performed using MLN. Given fixed values for the action nodes, LightLocation and Intensity; the inference will give the probability for each grounding of the utility predicates, RightArea and Comfort. Let's consider the case where action nodes are fixed as $a = (LightLocation(kitchen), Intensity(low))$, then for this configuration we obtain the following expected utility:

$$EU(a) = \sum_{x \in \{bad, acceptable, good\}} P(RightArea(x) \mid a).U(RightArea(x)) + \sum_{x \in \{low, medium, high\}} P(LightLocation(x) \mid a).U(LightLocation(x))$$

The optimal assignment of actions will be the one having the maximal EU.

6 Discussion and Future Work

Dealing with context in pervasive environments involves treating uncertainty, imprecision, and incompleteness; and so far, not a single method can overcome all these problems. Therefore Ambient Intelligence projects must rely on the application of several methods sharing a common base and serving each one a specific purpose. Our proposed framework is an attempt towards this direction.

Decision making by means of Markov logic networks seems very promising as it can take the best of logic and probabilistic models: a simple and clear representation in the framework while being able to treat uncertainty through probabilistic inference. However, as most of probabilistic models, MLN learning requires a considerable amount of data to estimate the optimal parameters. Unfortunately, corpora on pervasive environments with annotated data useful for decision making is rarely available. Furthermore, to the best of our knowledge there is no available corpora for decision making from vocal orders. Therefore, we plan, in the short term, to carry out experiments on a real SH platform that will provide us with data to optimize our decision models and to test the complete framework in realistic circumstances.

To further improve our framework, we intend to work on two improvements: the first one relates a tighter integration of the decision model with the ontology. We consider very interesting the possibility to check for coherence of the decision model rules by means of an ontology reasoner. In general, this integration is not trivial as MLN rules are defined in first-order logic, while description logic and safe rules are only a subset of first-order logic. Our second idea consists in extending the semantics of the situation recognition module in order to be able to define situations in terms of complex events.

References

1. Pedro Chahuara, Anthony Fleury, Michel Vacher, and François Portet. Méthodes SVM et MLN pour la reconnaissance automatique d'activités humaines dans les habitats perceptifs: tests et perspectives. In *Actes de la conférence RFIA 2012*, pages 340–347, Lyon, France, jan 2012.
2. Pedro Chahuara, François Portet, and Michel Vacher. 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*, 2011.
3. Liming Chen and Chris Nugent. Ontology-based activity recognition in intelligent pervasive environments. *International Journal of Web Information Systems*, 5(4):410–430, November 2009.
4. Berardina De Carolis and Giovanni Cozzolongo. C@sa: Intelligent home control and simulation. In Ali Okatan, editor, *International Conference on Computational Intelligence*, pages 462–465. International Computational Intelligence Society, 2004. International Conference on Computational Intelligence, ICCI 2004, December 17-19, 2004, Istanbul, Turkey, Proceedings.
5. Tao Gu, Hung Keng Pung, and Da Qing Zhang. A service-oriented middleware for building context-aware services. *Journal of Network and Computer Applications*, 28(1):1–18, January 2005.

6. P. Hitzler and B. Parsia. Ontologies and rules. In S. Staab and R. Studer, editors, *Handbook on Ontologies*, pages 111–132. Springer, Heidelberg, 2nd edition, 2009.
7. R. Howard and J. Matheson. Influence diagrams. *Readings on The Principles and Applications of Decision Analysis*, 1 and 2:720, 1981.
8. Michael Klein, Andreas Schmidt, and Rolf Lauer. Ontology-centred design of an ambient middleware for assisted living: The case of soprano. In *30th Annual German Conference on Artificial Intelligence (KI 2007)*, 2007.
9. Hsien-Chou Liao and Chien-Chih Tu. A rdf and owl-based temporal context reasoning model for smart home. *Information Technology Journal*, 6(8):1130–1138, 2007.
10. Daniel Lowd and Pedro Domingos. 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, 2007.
11. A. Mileo, D. Merico, and R. Bisiani. Reasoning support for risk prediction and prevention in independent living. *Theory Pract. Log. Program.*, 11(2-3):361–395.
12. Aniruddh Nath and Pedro Domingos. A language for relational decision theory. In *Proceedings of the Sixth International Workshop on Statistical Relational Learning*, Leuven, Belgium, 2009.
13. Aniruddh Nath and Pedro Domingos. Efficient belief propagation for utility maximization and repeated inference. In *AAAI*, 2010.
14. Takashi Nishiyama, Shinpei Hibiya, and Tetsuo Sawaragi. Development of agent system based on decision model for creating an ambient space. *AI & Society*, 26(3):247–259, 2011.
15. Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine Learning*, 62(1-2):107–136, 2006.
16. Bill Schilit, Norman Adams, and Roy Want. Context-aware computing applications. In *In Proceedings of the Workshop on Mobile Computing Systems and Applications*, pages 85–90. IEEE Computer Society, 1994.
17. Son D. Tran and Larry S. Davis. Event modeling and recognition using markov logic networks. In *Proceedings of the 10th European Conference on Computer Vision: Part II*, pages 610–623, Marseille, France, 2008. Springer-Verlag.
18. Michel Vacher, Dan Istrate, François Portet, Thierry Joubert, Thierry Chevalier, Serge Smidtas, Brigitte Meillon, Benjamin Lecouteux, Mohamed Sehili, Pedro Chahuara, and Sylvain Méniard. The sweet-home project: Audio technology in smart homes to improve well-being and reliance. In *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2011)*, Boston, USA, 2011.
19. Peter Wolf, Andreas Schmidt, and Michael Klein. 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*, 2008.