Making Context Aware Decision from Uncertain Information in a Smart Home: A Markov Logic Network Approach

Pedro Chahuara, François Portet, Michel Vacher

Making Context Aware Decision from Uncertain Information in a Smart Home: a Markov Logic Network Approach

Pedro Chahuara, François Portet, and Michel Vacher *

Laboratoire d'Informatique de Grenoble
Grenoble 1/Grenoble INP/CNRS UMR 5217, F-38041 Grenoble, 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 focus of this paper is on the context-aware decision process which uses a dedicated Markov Logic Network approach to benefit from the formal logical representation of domain knowledge as well as the ability to handle uncertain facts inferred from real sensor data. The approach has been experiemented in a real smart home with naive and users with special needs.

Keywords: Sensing and Reasoning Technology, Knowledge-Based Systems, Decision making, 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 real cases context is inferred from sources affected by uncertainty.

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. For instance, the Open AAL platform [25] 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

* 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)

by logical reasoners, as the only purpose of the ontology is to be an artefact of integration. Chen et al. [4] 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 [15], 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. [17] 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 [7]. Influence diagrams [10], which are based on Bayesian networks, have been also applied to model the causal relation among decision actions, uncertain variables, risk, and utilities [20, 5]. 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. In this paper, we propose an approach involving the representation of concepts by means of ontologies and a set of logical rules. 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 SH context in Section 2 and the framework in Section 3. Section 4 details the decision making model and Section 5 summarises experiments conduced in a real smart home. Finally, a brief discussion is given Section 6.

2 The Smart Home context

The typical smart homes considered in the study are the one that permit voice based interaction. There is a rising number of such smart homes [11, 2, 8, 6, 14] that are particularly adapted to people in loss of autonomy [21]. Typically, such smart homes are multi-rooms and equipped with sensors and actuators such as infra-red presence detectors, contact sensors... This kind of smart home can support daily life by taking context sensitive decisions based on the current situation of the user. More specifically, the smart home can be *reactive* to vocal or other commands to make the most adequate action based on context, and can act *pro-actively* by recognising a specific situation in which an action must be made (e.g., for security issue). To illustrate this support, let's consider the following two scenarios:

> **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 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.*

> **Scenario 2** *The inhabitant returns to her apartment 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 dangerous 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.*

From these scenarios it can be noticed that contextual information, such as location and activity, play a major role in delivering the appropriate support to the user. In this paper, Location and Activity are defined 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. 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 $A : t_b, t_e \to a$, $t_b, t_e \in \mathbb{N}$ and $t_b < t_e$*

Moreover, much 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 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 $C_i^{k_i}$ establish a set $D_i \subset DOM(V_{k_i})$ to constrain 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 we have : V_1, V_2 and $V_{3,\ldots,n}$, which are the states of the main door, the user's location and the states 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: $D_1 = \{open\}, D_2 = \{\neg kitchen\}, D_{3,\ldots,n} = \{off\}$. 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 between S_k^1 and S_k^2.*

Consider $T_1 =< S^1, S^2, r >$ 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 can also be a 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 [1]. 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 informations characterizing the circumstance under which an inference is made.*

The main usage of context is disambiguation. When a situation is recognized, several decisions can be made with different effects. The context provides the complementary information to evaluate the circumstance in terms of risk (safety) 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.*

Though risk definition varies according to the domains, in decision making, risk is often a consequence of uncertainty which is evaluated by enumerating all the possible outcomes with their probability 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 decided by the decision making system.*

Under uncertainty, an action can have numerous effects. If the effect leads to a negative outcome, U takes a negative value. There is thus a relationship between U and the risk: to compute risk for a given action, the probability of all the unwanted states (i.e., those with a negative value) must be computed.

For instance, in Scenario 1 the situation which triggers the decision making is the recognition of the voice order *"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 light on the ceiling light or the bedside lamp or both. The effects could be do 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.

It must be emphasized that the choice depends on the context. Indeed, in the case of Scenario 1, as the person has just awoken in the dark, the bedside lamp would be the best choice to avoid dazzling, but in a different context (e.g., when tidying up) the ceiling could be the best choice.

3 The Voice Controlled Smart Home System

The smart home system we are considering in the study has been developed in the Sweet-Home project[24]. The reasoning capabilities of the system are implemented in the intelligent controller depicted in Figure 1. The bottom of the figure shows external systems that are connected to the controller to gather streams of data and send orders to the home automation system. All these streams of information are captured and interpreted to recognize situations and makes decisions.

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 and the current activity. These informations are useful to reduce ambiguity in the decision making process. Other works have also reckoned location and activity as fundamental for context-aware inference [17, 23]. In

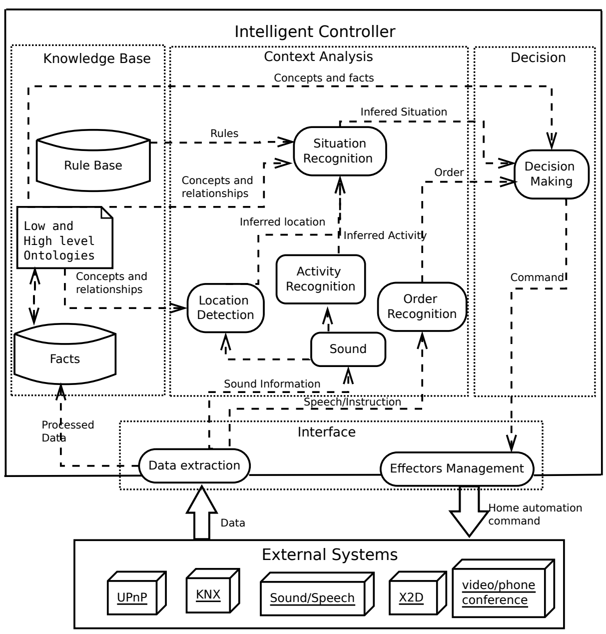

Fig. 1. The Intelligent Controller Diagram.

order to perform location and activity inference, two independent modules were developed and integrated in the framework. The former applies a method based on the modelling of the links between sensor events and location assumptions by a two-level dynamic network. Data fusion is achieved by spread activation on the dynamic network. The second module uses a classifier, based on Markov logic networks, to carry out activity recognition. Due to space limitation the reader is referred to [3] for further details.

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 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 modules. Within the controller architecture, other inference modules can be added without compromising the processing of the other components.

The knowledge of the controller is defined using two semantic layers: the *low-level* and the *high-level* ontologies. The two ontologies were implemented in OWL2, 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 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 makes possible a higher re-usability of the reasoning layer when the sensor network and the home must be adapted [13].

As situation are defined as temporal patterns of the system state, ontologies provide an appropriate foundation for situation recognition since they store all the facts (i.e., the system state) 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 [9]. These rules are built using the Semantic Web Rule Language (SWRL). However, the scope of this approach is very limited as it does not al-low to specify complex definitions. But, 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 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 her windows can be described by the SWRL rule 1 while the situation in scenario 2 can be modelled by rule 2.

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),curret_state(?a,on), swlrb:moreThan(sqwrl:count(?w), 1), → current_state(BedroomWindowsOpen, detected)

Rule 2 DeviceEvent(?l), has_associated_object(?l, light), takes_place_in(?l, bedroom), state_value(?d, off), Window(?w), located_in(?w, bedroom),Application(?a1), has_application(?w,?a1),curret_state(?a1,on), swlrb:equals(sqwrl:count(?w), 0), Blind(?b), located_in(?b, bedroom),Application(?a2), has_application(?b,?a2),curret_state(?a2,on), swlrb:equals(sqwrl:count(?b), 0), Door(?d), located_in(?d, kitchen),Application(?a3), has_application(?bd,?a3),curret_state(?a3,on), swlrb:equals(sqwrl:count(?d), 1) → current_state(MainDoorOpen, detected)

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

4.1 Modelling the decision making by Influence Diagrams

Influence diagrams [10] 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. Finally, utility nodes 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 A and an assignment of choices $a \in A$, 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 $a_{best} = argmax_a EU(a)$.

Figure 2 shows an example of Influence Diagram, based on the scenario 1. In this case, the setting of action variables, represented by rectangular nodes, indicate 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 of the inhabitant. Note that this location is not easy to determine in some cases since the inhabitant could be moving in the SH 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 of 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.

4.2 Markov Logic Networks (MLN)

MLN [22] 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 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.

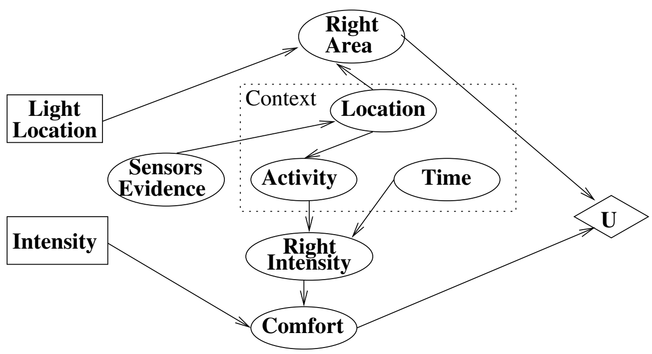

Fig. 2. Influence diagram for a decision after a vocal order is recognised.

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 (in our case, input data). During the inference process [22], every MLN predicated is 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 2.

$$P(X = x) = \frac{1}{Z} exp\left(\sum_{f_i \in F} w_i n_i(x)\right) \tag{2}$$

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 ith clause in the possible world x.

Because computing Z involve 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 [22].

MLN models can be acquired by supervised learning which 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 (2), the derivative of the log-likelihood wrt the weight w_i is given by equation (3).

$$\frac{\partial}{\partial w_i} \log P_w(X = x) = n_i(x) - \sum_{x'} P_w(X = x') n_i(x) \tag{3}$$

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, the computing equation (3) is intractable in most cases. Thus, approximation method are used in practice such as the *Scaled Conjugate Gradient* [16].

Since a Markov network is more general than a Bayesian network, Influence diagrams can also be implemented by means of MLN [18]. Nath et al. [19] 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: Firstly, MLNs are 1st order 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 2 can be defined as follows:

Predicate	**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

Weight	**Rule**
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	$Activity(a) \wedge Agitation(a, degree) \wedge Intensity(d) \rightarrow Comfort(high)$
-0.79	$Activity(a) \wedge Agitation(a, d1) \wedge Intensity(d2) \wedge d1! = d2 \rightarrow Comfort(medium)$
-0.09	$Activity(a) \wedge Agitation(a, d1) \wedge Intensity(d2) \wedge d1! = d2 \rightarrow Comfort(low)$

Utility Value

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

Evidence

NextTo(kitchen,bedroom)	NextTo(bedroom,study)	Agitation(rest,low)
Agitation(sleep,low)	Agitation(eat,low)	Agitation(tidy,high)
Agitation(hygien,high)	Agitation(dress up,high)	Agitation(communication,high)

This MLN is a template for constructing Markov network modelling an influence diagram. It must be constructed dynamically since the probability of context variables, location and activity, can not be known *a priori*. As shown Figure 3, once the decision module is triggered, it gets the evidences from the ontology instances that are used to ground the MLN and generates an influence diagram (actually a Markov network). This grounded network is then used to compute the action that maximize the expected utility using equation 4.

$$\begin{aligned} EU(a) = & \sum_{x \in \{bad,fair,good\}} P(RightArea(x) \mid a).U(RightArea(x)) \\ & + \sum_{x \in \{low,med.,high\}} P(LightLocation(x) \mid a).U(LightLocation(x)) \end{aligned} \tag{4}$$

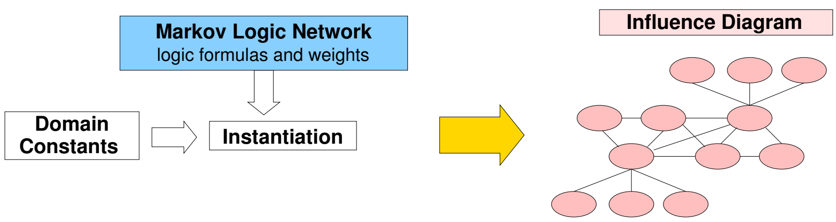

Fig. 3. Influence diagram construction by MLN grounding.

4.3 Making decision with uncertain information

As presented in Figure 1, contextual information, such as location and activity, results from an inference process. As such, contextual 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 [12] 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 5:

$$P'(X) = \textstyle\sum_{i=1}^{n} P(X \mid Activity_i, a).P(Activity_i) \qquad (5)$$

From equations 1 and 5, EU can then be estimated by equation 6. Note that $Activity$ is no more included in the set of contextual evidence e.

$$EU(a) = \sum_{X}\sum_{i=1}^{n} P(X \mid Activity_i, a, e).P(Activity_i).U(X) \qquad (6)$$

5 Experiments

The method was experimented in real situations in a smart home with 'typical' naive users and users with special needs interacting with the environment. This section describes the experimental set up and the results of the decision making for the 'typical' users and some preliminary feedbacks from the users with special needs.

5.1 Experimental set up and collected data

The experimental smart home is depicted Figure 4. It is a $32m^2$ flat including a bathroom, a kitchen, a bedroom and a study, all equipped with sensors and effectors such as infra-red presence detectors, contact sensors, video cameras (used only for annotation purpose), etc. In addition, seven microphones were set in the ceiling. The technical architecture of Domus is based on the KNX bus system, a worldwide ISO standard (ISO/IEC 14543), and include several other field buses as well, such as UPnP (Universal Plug and Play) for the audio video distribution, X2D for the opening detection (doors, windows, and cupboards), etc. More than 150 sensors, actuators and information providers are managed in the flat.

The experiments consisted in following a scenario of activities without constraints of time or the way of performing them. During the scenario, the participants had to utter several voice commands. A previous visit was organised

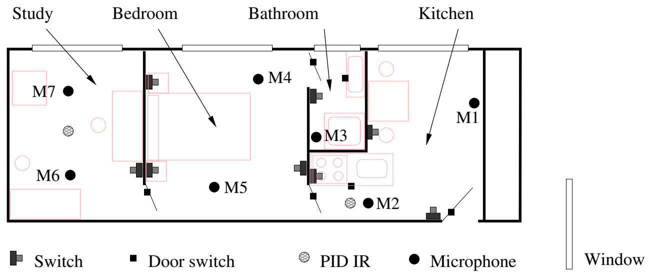

Fig. 4. The DOMUS Smart Home

so that the participants find all the items necessary to perform the activities. Many decisions were to be made by the decision module such as answering orders related to giving the time or closing the blinds. Due to space limitation, we will focus the paper on context aware decisions. In this case, 4 situations were specifically considered when the user utters the voice order *"turn on the light"*. In each situation, 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.

Moreover, the two situations described in Section 1 related to forgetting to close a window or the front door where included in the scenario. Each time these situation were recognized, a warning message was generated.

15 persons (including 9 women) participated to the experiment to record sensors data in a daily living context. The average age of the participants was 38 ± 13.6 years (19-62, min-max). At the end of the study, 11 hours of data was recorded (50 minutes per experiment in average).

5.2 Results of the decision making

Despite the time devoted to the experiment, the dataset was insufficient to learn the weights of a MLN decision model. Thus, the training corpus for weight learning was the result of the simulation of 200 instances, most of which expressing the best location and intensity but also including contradictory configurations. For instance, if in most of the situation 1 cases, the best light is the one above the table, a ceiling light can also be acceptable and very rarely the one above the table is considered as a bad decision. The learned weights are the one of the model presented in Section 4.2. From these weights it can be understood that

Target/Hit	Eat	Tidy	Dress	Sleep	Rest	Hygiene	Talk
Eat	9	6	0	0	0	0	0
Tidy	3	20	1	1	4	0	1
Sleep	1	2	1	10	1	0	0

Table 1. Confusion matrix of the activity recognition during decision making

the best location is always preferred while an incorrect intensity is not a high risk for the comfort.

Despite the scenario, the participants took some liberty and in some cases the warning situations were not recognized. The 15 instances of the warning situation 1 and 2 were recognized 8 and 5 times respectively. For each recognized situation, the intelligent controller acted immediately to deliver an appropriate warning message.

As discussed before, activity recognition is a difficult task which deliver uncertain information. In this paper, we focus strictly on the activity recognition during a voice command whose performance is presented in Table 1. There were 60 activity instances performed during voice command, they belong to: sleeping, eating and tidying up. However, our model has been trained to recognize seven activities (see [3] for more details). The most important confusion is between eating and tidying up. Both activity are performed in the kitchen and share many characteristics such as the noise produced by the dishes. The overall accuracy is of 65% which 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.

Table 2 shows the overall correct decision rate for each situation. 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 6. In practice, the uncertainty of the location was close to 100%, thus the uncertainty was mainly due to the activity recognition.

The worst performance is in the situation 1. This is mainly due to the confusion between eating and tidying up. However, the tidying up activity was well recognized and this explain the high accuracy for situation 2. Overall, the results with and without uncertainty are very close. They actually differs in only 5 instances out of 60. For instance, in the situation 3 for the participant 12, the

Situation/Expected Utility	without uncertainty	with uncertainty
Situation 1	54%	54%
Situation 2	93%	100%
Situation 3	73%	86%
Situation 4	60%	53%
Total	70%	73%

Table 2. Correct decision rate with and without activity uncertainty

activity recognition output was : hygiene(0.20), dressing (0.16), sleeping (0.28), and resting (0.17) while the ground truth was tidying up (0.08) in the kitchen [1]. 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.

5.3 Preliminary Results from experiments with the aged and visually impaired population

The method has also been applied in the same context but with aged and visually impaired people. The aim was both to validate the technology with this specific population and to perform a user study to assess the adequacy of this technology with the targeted users. In this experiment, eleven participants, either aged (6 women) or visually impaired (2 women, 3 men), were recruited. The participants were asked to perform 4 scenarios involving daily living activities and distress or risky situations. The average age was 72 years (49-91, min-max). During this experiment, 4 hours and 39 minutes of data was collected including the same sensors as the one previously described. All the participants went through a questionnaire and a debriefing after the experiment. At the time of writing, we are still analysing the results but overall, none of the aged or visually impaired persons had any difficulty in performing the experiment. They all appreciated to control the house by voice.

6 Discussion and Future Work

Dealing with context in pervasive environments involves treating uncertainty, imprecision, and modeling complex relational information; 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. The system we developed integrates several components that are devoted to specifics aspects of a smart homes. Thus, the whole framework covers the requirements of expressibility and uncertainty treatment.

Decision making by means of Markov logic networks presents many advantages. First of all, MLN relying on a formal logic representation which is particularly suited to Ambient Intelligent systems where knowledge is often represented by means of logic. When possible, this permits translation from one representation to another to perform, for instance, addition of relational knowledge as expert knowledge in the MLN structure learning. In this perspective, the use of a formal domain knowledge description and logic-based decision method could lead to a higher re-usability of the model in another smart home. Secondly, MLN,

[1] 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 activity and an hygiene activity can both have a high probability

being a probabilistic model, can deal with uncertainty and make inference from a incomplete input.

However, as most of probabilistic models, MLN 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. 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. 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.

The experiments carried out in a real SH platform with naive and targeted users has shown that our approach is promising both regarding decision making and the overall system. From this research, many studies can be conducted to improve the decision making. Given that decision data have been acquired, the a priori model could feed a learning with this reduced set in order to adapt the model to the specific home environment. Furthermore, information is uncertain in the smart home environment, thus the handling of the uncertain evidence must be generalised. Regarding knowledge representation, a tighter integration of the decision model with the ontology would be desirable. 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.

References

1. Alexander Artikis, Anastasios Skarlatidis, François Portet, and Paliouras G. Logic-Based Event Recognition. *Knowledge Engineering Review*, 27(4):469–506, 2012.
2. A Badii and J Boudy. CompanionAble - integrated cognitive assistive & domotic companion robotic systems for ability & security. In *SFTAG 2009*, pages 18–20, Troyes, France, 2009.
3. 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.
4. 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.
5. Berardina De Carolis and Giovanni Cozzolongo. C@sa: Intelligent home control and simulation. In *International Conference on Computational Intelligence*, pages 462–465, 2004.
6. G. Filho and T.J. Moir. 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, 2010.
7. 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.

8. M Hamill, V Young, J Boger, and A Mihailidis. Development of an automated speech recognition interface for personal emergency response systems. *Journal of NeuroEngineering and Rehabilitation*, 6, 2009.
9. 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.
10. R. Howard and J. Matheson. Influence diagrams. *Readings on The Principles and Applications of Decision Analysis*, 1 and 2:720, 1981.
11. Dan Istrate, Michel Vacher, and Jean-François Serignat. Embedded implementation of distress situation identification through sound analysis. *The Journal on Information Technology in Healthcare*, 6:204–211, 2008.
12. Richard C. Jeffrey. *The Logic of Decision*. University Of Chicago Press, July 1990.
13. 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.
14. Benjamin Lecouteux, Michel Vacher, and François Portet. Distant Speech Recognition in a Smart Home: Comparison of Several Multisource ASRs in Realistic Conditions. In *Interspeech 2011*, pages 2273–2276, Florence, Italy, aug 2011.
15. 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.
16. Daniel Lowd and Pedro Domingos. Efficient weight learning for markov logic networks. In *Proceedings of the Eleventh European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 200–211, 2007.
17. A. Mileo, D. Merico, and R. Bisiani. Reasoning support for risk prediction and prevention in independent living. *Theory and Practice of Logic Programming*, 11(2-3):361–395, 2011.
18. Aniruddh Nath and Pedro Domingos. A language for relational decision theory. In *International Workshop on Statistical Relational Learning*, Leuven, Belgium, 2009.
19. Aniruddh Nath and Pedro Domingos. Efficient belief propagation for utility maximization and repeated inference. In *AAAI*, 2010.
20. 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.
21. François Portet, Michel Vacher, Caroline Golanski, Camille Roux, and Brigitte Meillon. Design and evaluation of a smart home voice interface for the elderly — Acceptability and objection aspects. *Personal and Ubiquitous Computing*, 17(1):127–144, 2013.
22. Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine Learning*, 62(1-2):107–136, 2006.
23. Bill Schilit, Norman Adams, and Roy Want. Context-aware computing applications. In *Proceedings of the Workshop on Mobile Computing Systems and Applications*, pages 85–90. IEEE Computer Society, 1994.
24. 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 *EMBC 2011*, Boston, USA, 2011.
25. 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*, Patras, Greece, 2008.