EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

A hybrid approach to inferring the Internet of Things for complex activity recognition

Qingjuan Li[1,2], Huansheng Ning[1,2*], Tao Zhu[3], Shan Cui[1,2] and Liming Chen[4]

Abstract

With the rapid development and large-scale uptake of the Internet of Things, smart home is evolving from a vision towards a realistically viable solution for assisted living. Activity recognition is one of the fundamental tasks in order to provide accurate and timely assistance and service. As daily living scenarios are full of similar activities, missing data, and noise, inferring complex activities using knowledge-driven reasoning algorithms suffers from several drawbacks, e.g., real-time raw sensor data segmentation, poor generalization, higher computational complexity, and scalability. To address these problems, this paper proposes a hybrid approach to complex daily activity recognition by merging the first-order logic and probability graphic modeling. Specifically, we develop a novel "Markov logic network" combining data-driven multi-feature and simplified rule-based modeling and inference, thus enabling and supporting the applicability and robustness of daily activity recognition. To evaluate the approach and associated methods, we design a testing scenario with a number of similar activity groups, missing data, or disturbance test datasets in a multi-modeling sensor scene. Initial results show our approach outperforms the traditional approach with a better accuracy in the situations of similar activities with missing data and noise disturbance. Experiments are also conducted to compare the Gibbs sampling and MC-SAT sampling algorithms for Markov logic network, and the results show that the Gibbs is better in our experimental settings.

Keywords: Sensors, Complex activity recognition, Markov logic network, Data missing or disturbance

1 Introduction

Activity recognition is a fundamental task for pervasive computing, context-aware computing, smart activity, healthy aging, wise information technology, and etc. [1]. In the literature of activity recognition, most of the works build activity model based on the temporal spatio characters [2, 3], which faces the miscalculation with the similar activities because of the few features. Another research bottleneck is the poor robustness of the activity model which is difficult to handle the data missing, noisy data, habit changing, and so on [4, 5]. In addition, the complex multi-activities are more popular which require an active partition method for the continuous time sequence sensor data [6–8]. Therefore, designing a comprehensive solution for the daily activity recognition is the main work of this paper.

Recognition of activity of daily living (ADL) aims to recognize the residents' action intent or intention (activity), living habits, and health status to detect the abnormal behavior and provide the personalized service. There are three types of devices to collect data: vision equipment, wearable devices [9], and ambient sensors. The wearable devices need to be worn all time and charged in time; the wearable devices are excluded in our research which sometimes cause interference. Vision and ambient sensors have their own merits and demerits respectively. The vision sensors have been widely adopted directly [10, 11] which is obvious to record in video. While the privacy requirements have been paid attention to gradually, ambient sensors are easier to be accepted by residents. Therefore, the non-vision ambient sensors have become the research hotspot [12]. That method also has another

*Correspondence: ninghuansheng@ustb.edu.cn
[1] School of Computer and Communication Engineering, University of Science and Technology Beijing, No.30, Xueyuan Road, Beijing, 100083 China
[2] China and Beijing Engineering Research Center for Cyberspace Data Analysis and Applications, No.30, Xueyuan Road, Beijing, 100083 China
Full list of author information is available at the end of the article

advantage that the multi-modeling sensing data has more proofs for the inference to make the more accurate result. Like human beings, the feeling of touching, smelling, tasting, and hearing and their attributes can be sensed and visible by sensors. For many home activities, ambient sensors like infrared, touch, pressure, and light are more intuitive to represent their related features. A typical indoor scene layout of multi-modeling sensors has been shown in Figs. 1 (kitchen), 2 (living room), 3 (water machine), 4 (cup model), and 5 (deployed cup model).

Raw data collection is the first step, and the second step is establishing the inference model. The usual inference methods adopted can be divided two categories, data-driven and knowledge-driven measures. The data-driven method trains the activity model which represents the implicit general rules by huge numbers of labeled data. Actually, the earliest studies have adopted the data-driven method mostly [13–15]. This method is a typical probability statistical model which has good performance on single resident's single activity recognition. For example, the well-known key technologies include hidden Markov models and dynamic Bayesian network. It belongs to the series probability graphical model which has the high accuracy and the ability of mining complex correlation for complex activities. But the model is usually trained by the experimenters who are not the users. There has an essence problem for the different distribution between test data and train data which means the model is not impracticable and inapplicable to recognize the personalized activities. Another method, knowledge-driven, is an expert system relying on the background and domain knowledge. This knowledge builds mapping from activity entity, appliance entity and context (sensor entity, sensor value entity), temporal-spatial traits, hierarchical methodology, and rules (formulas) [16, 17]. This method has shown a good performance in using directly to overcome "cold start" problem [18]. The general rules of the model represent most residents' activity habits which have a high reusability, while the preference, details, and sufficient description of one activity cannot be taken into full consideration. For a better performance, the activity model is dynamic and uncertain which should be updated with the habits changing. How to match best results? Try to combine the two methods to learn from other's strong points to offset one's weakness which is called the hybrid model [19]. Gabriele proposed a novel hybrid approach with probabilistic and knowledge-based reasoning and adopted the unsupervised feedback to label the rules which have good performance in the real-world dataset [20]. Shin et al. presented the regular expression-based string matching algorithm; the pattern model is trained by labeled data based on the experts' knowledge base [21]. These hybrid methods have good performance for complementing and correcting each other. Expert knowledge is more consistent which is utilized at the first step to model the basic formula [22]. Data modification is more flexible which

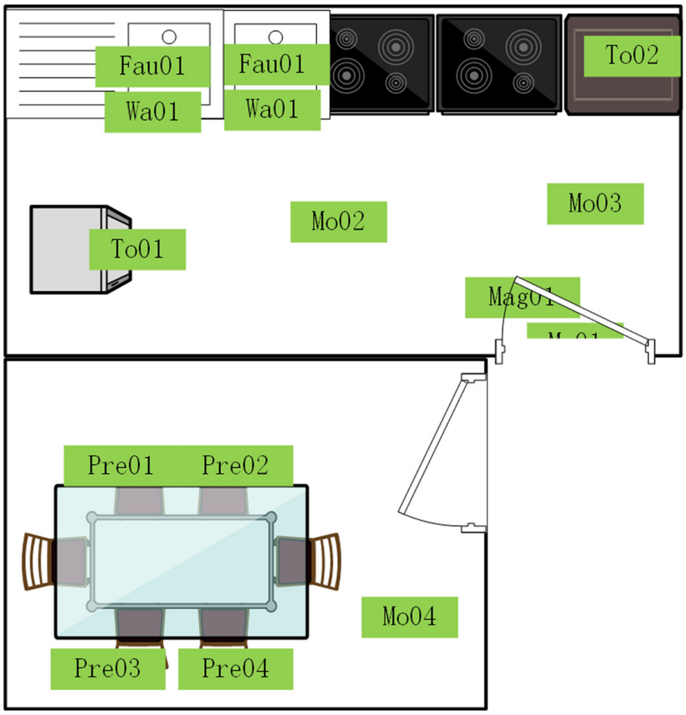

Fig. 1 A kitchen layout of multi-modeling sensors

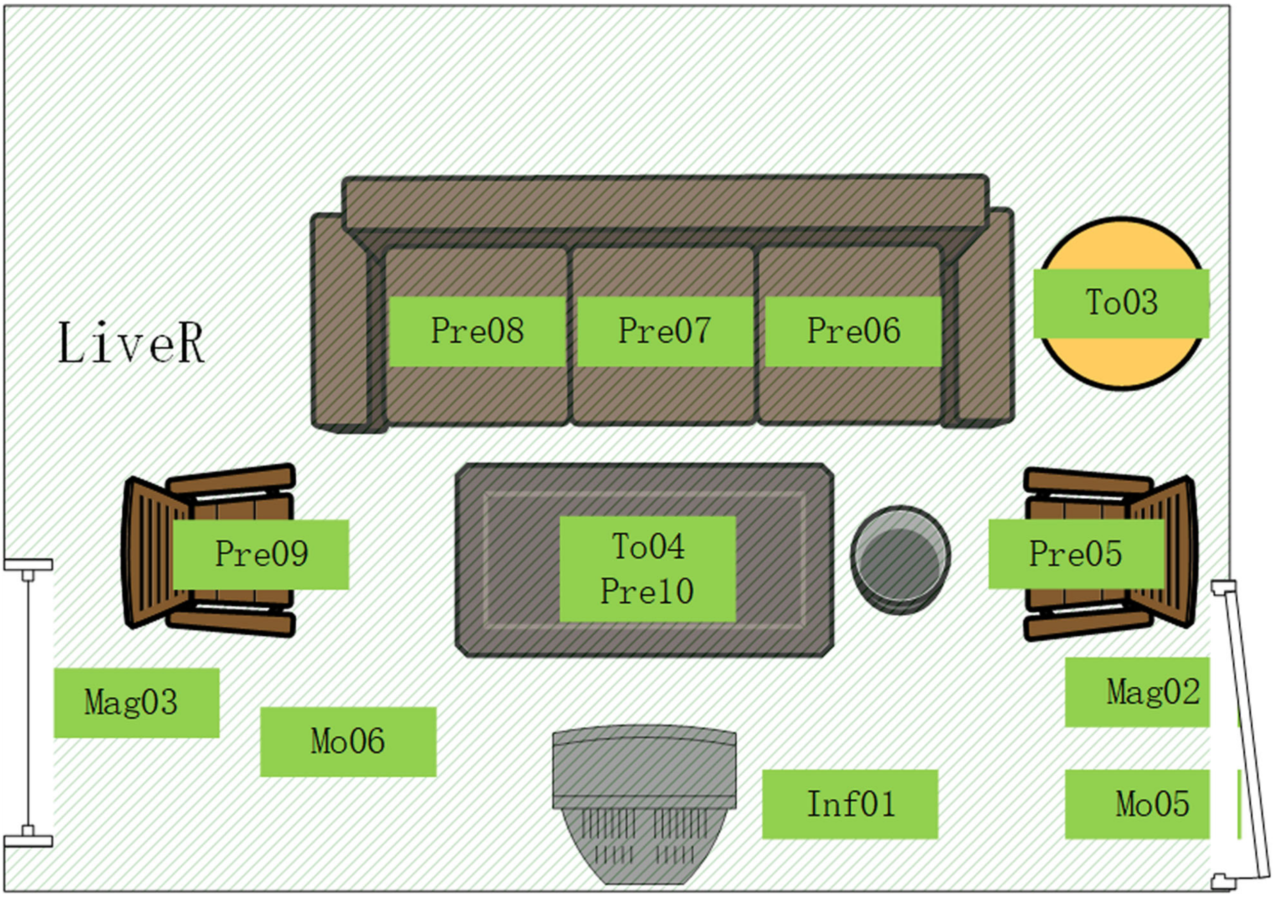

Fig. 2 A living room layout of multi-modeling sensors

revises the formula. In this paper, we adopt one hybrid model, Markov logic network (MLN), which uses both of the first-order logic (FOL) and statistical probability. Establish the rules' model by expert knowledge, then give the satisfiability probability to every rule by learning from series of real raw sensor data sequences. The main advantages are the the strong ability for complex activities and having the good performance to handle the dynamic and the data missing. The details are shown in Section 3.

The main contributions of our paper are as follows:

- Improving the Markov logic network by combining the simple words which can express the action, entity, time, time period, and place, which covers all attributes of sensor data and reduces the description complexity
- In order to improve the robustness of the activity model, we adopt the subrules to express the activity which just has one evidence for the result. Inference is a combination of the rules which is related to the activity

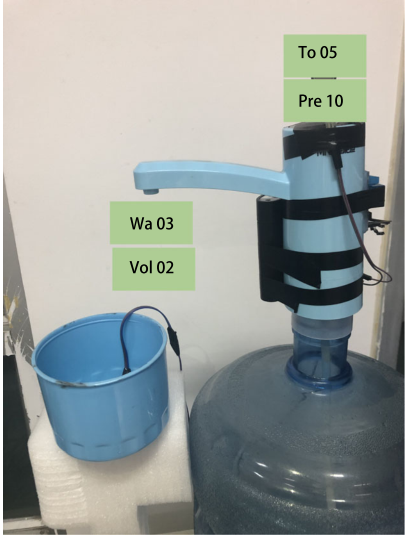

Fig. 3 A water machine model of multi-modeling sensors

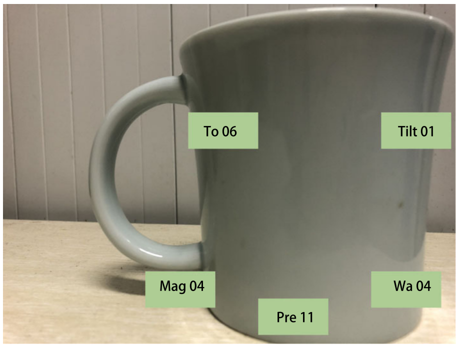

Fig. 4 A cup model of multi-modeling sensors

- Proving the Gibbs sampling algorithm has a better performance than the traditional adopted method, MC-SAT algorithm, in these complex activities

In this paper, Section 2 presents the background, Markov logic network, which merges the first-order logic and Markov network. A novel method based on the Markov logic network has been shown in Section 3, which includes data expression, entity expression, activity expression with time series and periods, and simplifying rule expression. Section 4 describes the experiment design and settings. Section 5 shows the results and discussion of different complex activity situations and contrasts the different algorithms in these special situations. Section 6 gives the conclusion and perspective.

2 Background (MLN)

2.1 Graphical model of the first-order logic

The first-order logic formula, also called as the first-order predicate calculus, consists of assertion and quantify. Constant symbols (e.g., time: 20181223 and 20190107) represent the real instance. Variable symbols (e.g., x and y) are the abstraction of constant symbols which range over the real instance. Predicate symbols (e.g., m UseBootle, UseTeabag) represent the relations and attributes mapping into a set of category events, including sensor events, entity events, and activity events. In addition, the object has its own attributes like location and usage which is one feature value to be added into layers. Term is any expression of object (constant, variable) (e.g., 20181223, x), like sensor or entity events, with the time character, for example, day-hour-minute-second. Atom is a predicate symbol applied to a tuple of terms or one term (e.g., UseTeabag(b)). Formulas are recursively constructed from atom using logical connectives and quantifiers, for example, negation ($\neg$), conjunction ($\wedge$), disjunction ($\vee$), implication ($\rightarrow$), and equivalence ($\leftrightarrow$) [23]. It is convenient to convert formulas to a more regular form, called clausal form which is defined by disjunctive normal form (DNF) [24]. Knowledge base is a conjunction of clauses, every clause is a disjunction of literals.

Graph *G* is composed by nodes and edges: nodes represent the terms, edges represent the relations, and clique of G represents first-order logic formulas (atoms). The formula is non-negative and real-valued. According to the graphical methodology, there is an edge between two nodes that appear in at least one formula.

2.2 Learning weight of the first-order logic by probability

First-order logic formula weights can be learned generatively by maximizing the likelihood of a relational training database. Automatically giving weight by data is much better than manually refining with less work. The learning algorithm is based on convex optimization. The gradient descent algorithm is one of the optimization methods for searching in first-order class which use the gradient *g*, with learning rate η to update the weight ω; the formula has been shown in (1) [25]. The formula of *g* has been shown in (2), where $E_{\omega,y}$ is the expectation over

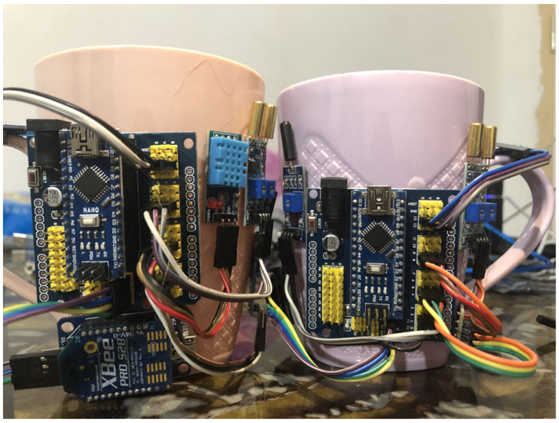

Fig. 5 A deployed cup model of multi-modeling sensors

the non-evidence atoms. For the second-order class, the optimization method aims at searching direction from function as a quadratic surface [26].

$$\omega_{t+1} = \omega_t - \eta g \tag{1}$$

$$\begin{aligned} g &= \frac{\partial}{\partial \omega_i}(-logP_\omega(Y=y|X=x)) \\ &= -n_i(x,y) + \sum_{y'} P_\omega(Y=y'|X=x)n_i(x,y') \\ &= E_{\omega,y}[\, n_i(x,y)] - n_i(x,y) \end{aligned} \tag{2}$$

2.3 A typical method-based first-order logic and probability (MLN)

Markov logic network (MLN) is a combination of the first-order logic and Markov networks [27]. Its knowledge model is first-order formulas with weights. The Markov network is one of the models for the joint distribution of a set of variables which is an undirected graph G and a set of potential function ϕ_k. Each state of $M_{L,C}$ (all nodes have values) represents a possible world.

The conception of MLN, L is a set of pairs (F_i, ω_i), with a finite set of constants $C = \{c_1, c_2, \ldots, c_{|C|}\}$, defines a $M_{L,C}$ (Markov network). In $M_{L,C}$, the nodes are binary value, 1 if the ground predicate is true, 0 otherwise. In $M_{L,C}$, each possible grounding of each formula has been represented. The fewer formulas a world violates, the more probable it is. The associated weight is bigger, reflecting the constraint is stronger.

The probability distribution over possible world x specified by the ground Markov network $M_{L,C}$ is given by formula (3) [27], where $n_i(x)$ is the number of the true groundings of F_i in x, $x_{\{i\}}$ is the state (truth values) of the predicates appearing in F_i, and $\phi_i(x_{\{i\}}) = e^{\omega_i}$ [28]. The second right part is the most convenient approach with a mixture of hard and soft constraints. Because for some formulas with 0 potential, it means the probability is 0.

$$\begin{aligned} P(X=x) &= \frac{1}{Z} exp\left(\sum_i \omega_i n_i(x)\right) \\ &= \frac{1}{Z}\prod_i \phi_i(x_{\{i\}})^{n_i(x)} \end{aligned} \tag{3}$$

Reasoning probabilities for complex relationships are the key step of this paper. Due to the size and complexity of the grounding Markov network, it is infeasible to inference many scenarios [29]. MLN has two inference types, one is the most likely state, another is the conditional probabilities. We adopt the second inference to calculate every ground query atom's conditional probabilities. Gibbs sampling is a typical algorithm of the Markov Chain Monte Carlo algorithm. The MC-SAT algorithm is a slice sampling Markov Chain Monte Carlo algorithm which combines the satisfiability testing and simulated annealing. Satisfiability solver (WalkSAT) is the efficiently finding isolated modes in the distribution [30].

All relevant atoms have been retrieved by the Markov blanket concept which has been mentioned before. The probability of a ground predicated X_l when its Markov blanket B_l is in state b_l is in (4). F_l is the set of ground formulas that X_l appears in, and $f_i(X_l = x_l, B_l = b_l)$ is the value of the ith ground formula when $X_l = x_l$ and $B_l = b_l$.

$$\begin{aligned} &P(X_l = x_l|B_l = b_l) = \\ &\qquad \frac{exp\left(\sum_{f_i \in F_l} \omega_i f_i(X_l = x_l, B_l = b_l)\right)}{exp\left(\sum_{f_i \in F_l} \omega_i f_i(X_l = 0, B_l = b_l)\right)} \\ &\qquad * \frac{1}{exp\left(\sum_{f_i \in F_l} \omega_i f_i(X_l = 1, B_l = b_l)\right)} \end{aligned} \tag{4}$$

MLN has many outstanding advantages in the following:

- Merging multiple expert knowledge even though these formulas are exclusive and inconsistent
- Learning and inferencing by ground atoms, even the not independent and identically distributed atoms
- Reducing the reliance on expert knowledge

3 Method

This paper offers a solution method based on MLN to build a recognition model from the bottom up, including the raw data pre-processing, expert system for knowledge rules, rules' weight learning by labeled data, and inference. The flow chart of the novel method has been shown in Fig. 6.

In Section 1, the processing approach for sensor data has been mentioned. In this section, the hierarchical concept graph will be explained in details in Fig. 7.

There are three layers, the first layer is converting the raw sensor data to an entity event. Sensors have been attached in physical objects and reflect an attribute of this object [31]. The entity event layer is established according to the activity concept. Activity is a series of entity sequences by single or multiple residents. An entity event is one action which has been triggered by a series sensor because the action must have closed connection with one physical entity.

3.1 Expression of sensor data

The collecting data of sensors has been stored as the following format. A data record has the sensor name, sensor id, sensor value, sensor triggered time, and sensor place.

$$Time, Place, SensorName, SensorID, Value \tag{5}$$

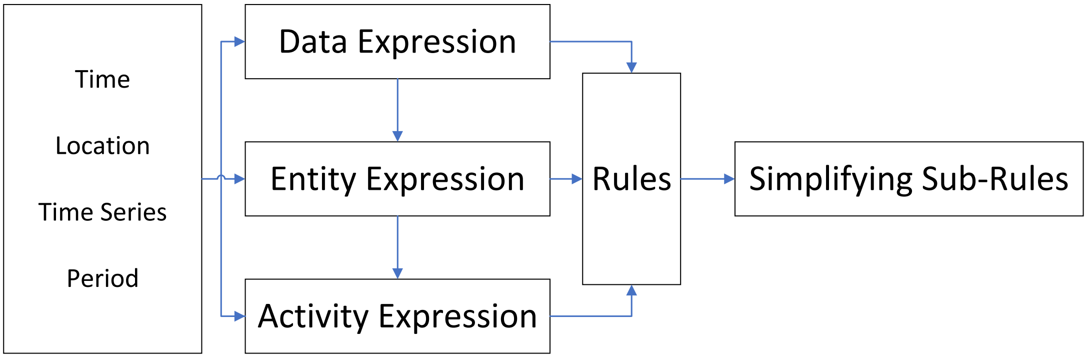

Fig. 6 The flow chart of the novel method

In order to recognize the similar activities, add the time period feature and time series to the data. Combine all attributes in one compact expression like:

$$\begin{cases} Sensor(ID)\_(Time, Place, TimePeriod), MeetsCondition \\ \neg Sensor(ID)\_(Time, Place, TimePeriod), otherwise \end{cases} \quad (6)$$

The specific definition of the sensor, place, and time period has been shown in Table 1, time including the year, month, day, hour, minute, and second.

3.2 Simplifying the activity rules by entity expression

The typical entity event is action events which consist of several continuous sensor data; multiple sensor events are triggered in a period of time. The related sensors can be combined together to generate the entity event which can be divided to three categories.

- Motion, touch, tilt, pressure,and light
  Action is one kind of the entity event, which leads to the position change or the relative position change of an entity. Using multiple kinds of sensors, including the human sensor and object sensor, closing to the object combining with the object self-change is the entity event which is triggered by a resident. The comprehensive rules of an entity event just concern a very short time of just several seconds and require the small range space to avoid a misunderstanding of the multi-resident.
- Pressure

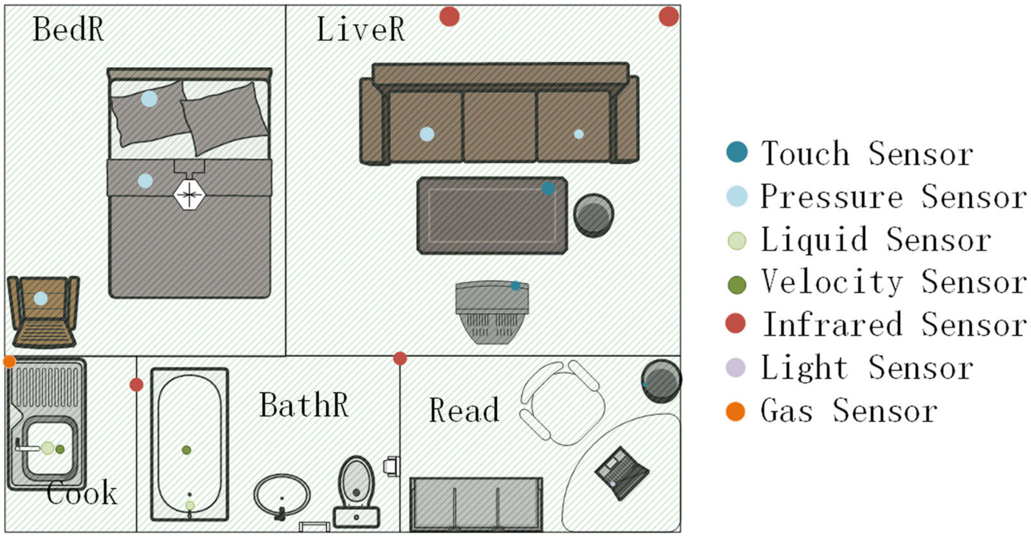

Fig. 7 Hierarchical methodology for sensor, entity, and activity

Table 1 Definition of the sensors, place, and time period

Sensor	Place	Time period
Motion	Bedroom	1,2,3,4
Touch	Living room	5,6,7,8
Light	Bathroom	9,10,11,12
Magnetic	Kitchen	0.5,1.5
Gas	Dining room	2.5,3.5
Water		4.5,5.5
Pressure		6.5,7.5
Tilt		8.5,9.5
Temperature		10.5,11.5
Humidity		

Weight change is also one kind of the entity event, which has the significant show for the drink, eat, and other weight transfer events obviously.

- Gas, temperature, and humidity
Environment change is key to the context awareness which is one kind of the entity event, when the environment parameter crosses the threshold value; the sensing data has been recorded which can provide assistance for other change situations.

Note that the time span of the entity event is just seconds which can be ignored, reducing the demand for storage and simplifying the inference process with the simple expression of evidence.

The entity event expression is shown in the following:

$$\begin{cases} Entity(ID)\_(Time, Place, TimePeriod), MeetsCondition \\ \quad \neg Entity(ID)\_(Time, Place, TimePeriod), otherwise \end{cases} \tag{7}$$

3.3 Activity expression

Activity events have more attributes than sensor and entity events, which add time period and change the triggered time to begin time and end time which is accurate to express the activities. The time period is assisting to distinguish similar activities which have similar entity event series. The evaluation and experiment have been shown in Section 4.

The activity event expression is shown in the following:

$$\begin{cases} Activity\_(Place, TimePeriod, BeginTime, EndTime), MeetsCondition \\ \quad \neg Entity\_(Place, TimePeriod, BeginTime, EndTime), otherwise \end{cases} \tag{8}$$

3.4 Time series expression

The time relationship has three kinds, *Before*, *After*, and *Equal*. There are some definitions of these relations.

- $Before(x,y) \leftrightarrow After(y,x)$
- $Before(x,y) \vee Equal(x,y) \leftrightarrow \neg Before(y,x)$
- $After(x,y) \vee Equal(x,y) \leftrightarrow \neg After(y,x)$
- $Equal(x,y) \leftrightarrow Equal(y,x)$

In the time relationship between the entity event (entity_time) and the activity event (activity_begintime, activity_endtime), we define that the activity begin time and end time have the relation as $Before(activity\_begintime, activity\_endtime)$. There were five kinds which are incompatible between each other, Before, After, Between, Begin, and End. The specific definition of them is shown in the following:

- $Before(entity\_time, activity\_begintime) \leftrightarrow Before(entity, activity)$
- $After(entity\_time, activity\_endtime) \leftrightarrow After(entity, activity)$
- $After(entity\_time, activity\_begintime) \wedge Before(entity\_time, activity\_endtime) \leftrightarrow Between(entity, activity)$
- $Equal(entity\_time, activity\_begintime) \leftrightarrow Begin(entity, activity)$
- $Equal(entity\_time, activity\_endtime) \leftrightarrow End(entity, activity)$

3.5 Simplifying rule expression (subrules)

References [32] and [33] have adopted MLN to recognize the activities, but they work mainly for sequence activities without the probability learning which is just the knowledge-driven method. Civitarese et al. have presented the duration concept by calculating the difference between the beginning time and ending time [20]. These papers design the multi-restricted conditions in one rule, but the satisfiability probability is for the whole rule which let the rejection for the rule just by a small dissatisfaction condition. But the dissatisfaction condition is generated by many situations, including the data missing, dynamic habits, and noisy data. In order to improve the flexibility of the activity model, the clausal form just includes the two event atoms which are group subrules of the traditional rules, one is an entity event, one is an activity event. While the time series of entity events also is one of the key evidence for inference, the time relationship between the entity event and activity event is also contained into the clausal.

For example, before we define the Drink activity rules as $UseCup(a)\_(x,z,p) \wedge DecreaseCup(a)\_(y,z,p) \rightarrow Drink\_(z,p,x,y)$, now we divide the rule to subrules as $UseCup(a)\_(x,z,p) \rightarrow Drink\_(z,p,x,y)$ and $UseCup(a)\_(x,z,p) \rightarrow Drink\_(z,p,x,y)$. The two evidence entity events have been consisted into different rules which can improve the flexibility of the inference.

4 Experiments

We designed three similar group (Drink Tea, Drink Coffee, Have Meal, Do Dishes, Sweep, Wipe) activities in kitchen and living room by a top-down approach in Table 2. In our experiment, two volunteers have been living in the environment 2 weeks (as the small training data set to improve the model), and label all the training data. Then, we have collected another 1-week data to test the model. The following data is based on the test data. From the bottom to top, we connect these sensors by Raspberry Pi, Arduino Mega 2560, Arduino Nano; communicate with PC by Serial Port Communication, Bluetooth HC-05, WIFI module, cellular network; and store the raw data in ".json" file. We segment the sensor data by the time window (width = 10 s), then obtain the entity event which has been stored into the ".db" file. Alchemy 2.0 is one of the engines to inference by MLN which we have used in this experiment.

Our work is discussing the performance (average precision rate) of the time period for similar activities, like Drink Coffee and Drink Tea. Another work is improving the robustness of the activity model with the missing data and error data. The last work is to compare the Gibbs sampling with MC-SAT and find the best method in complex activity situations.

5 Results and discussion

5.1 Similar activities by multi-attributes

For example, the terms "definition" and "first-order logic of *DrinkTea* and *DrinkCoffee*" have been shown in the following:

Table 2 Activity-entity-sensor design for five activities

Activity	Entity	Sensor
Sweep	Usefaucet	Rotate, velocity, water
	Usemop	Touch
Wipe	Usefaucet	Rotate, velocity, water
	Useduster	Touch, pressure
Drink (coffee, tea)	Usecup	Touch, tilt
	Useteabag	Touch
	Usecoffeebag	Touch
	Usebottle	Touch, pressure
Dishes	Bowl	Touch, pressure
	Chopstick	Touch, triaxial transducer
	Usefaucet	Rotate, velocity, water
Eat	Bowl	Touch, pressure
	Chopstick	Touch, triaxial transducer

//**Evidence**

$$\begin{aligned}&UseTeabag(id)\_(time, place, period)\\&UseCup(id)\_(time, place, period)\\&UseBottle(id)\_(time, place, period)\\&UseCoffeebag(id)\_(time, place, period)\end{aligned} \tag{9}$$

//**Query**

$$\begin{aligned}&DrinkTea\_(place, period, begintime, endtime)\\&DrinkCoffee\_(place, period, begintime, endtime)\end{aligned} \tag{10}$$

//**Rules**

$$\begin{aligned}&DrinkCoffee\_(p,l,x,z)\wedge!\,Before(x,z)\wedge!\,Before(z,y)\\&\wedge!\,UseCup(a)\_(x,p,l)\wedge!\,UseBottle(b)\_(z,p,l)\\&\wedge!\,UseCoffeebag(c)\_(y,p,l)\\&DrinkTea\_(p,l,x,z)\wedge!\,Before(x,z)\wedge!\,Before(z,y)\\&\wedge!\,UseCup(a)\_(x,p,l)\wedge!\,UseBottle(b)\_(z,p,l)\\&\wedge!\,UseTeabag(c)\_(y,p,l)\end{aligned} \tag{11}$$

Definition 1 *The term of the formula has two classes: variable uses all lowercase letters (e.g., x, y) and constant uses all uppercase letters or number letters (e.g., Bob, 20181225).*

Definition 2 *The predicate or atom names begin with uppercase letters (e.g., Drink, Work).*

Definition 3 *The evidence term and query term need to be noted before the inference.*

Definition 4 *The rules define by the temporal-spatial characteristic and the basic traits, the sequence has been realized by like Before predicate.*

Description 1 In the first-order logic, the negation by ¬ symbol, in MLN, is same the ! symbol.

In the experiment, the two activities have more attributes of the time series and time period and the accuracy of similar activities has been improved which is shown in Table 3. Contrasting with the time series and time period rules and traditional rules, the accuracy of similar activities has a significant improvement.

5.2 Missing data and disturbance situations

In this part, simplifying the rules by dividing the rules to the common rules and special rules are two parts. Because of the vital role of the special rules, initial weight value is bigger than common rules, the connection weight

Table 3 Two similar activities' probability

Activity group	Preci. (no)	Preci. (with series and periods)
DrinkTea and *DrinkCoffee*	0.025	0.910
HaveMeal and *DoDishes*	0.011	0.992
Sweep and *Wipe*	0.025	0.975

value between common and special rules is the same with the common rules. We have tested the missing data for different activities, and the accuracy has a significant improvement which has been shown in Table 4.

For example, the simplifying rules of *DrinkTea* and *DrinkCoffee* have been shown in the following:

$$\begin{array}{l}
//\textbf{SimplifyingRules}\\
DrinkCoffee\_(p,l,x,z)\wedge!\,Before(x,z)\wedge!\,UseCup(a)\_(x,p,l)\\
DrinkCoffee\_(p,l,x,z)\wedge!\,Before(x,z)\wedge!\,UseBottle(b)\_(z,p,l)\\
DrinkCoffee\_(p,l,x,z)\wedge!\,Before(x,z)\wedge!\,Before(z,y)\\
\wedge!\,UseCoffeebag(c)\_(y,p,l)\\
DrinkTea\_(p,l,x,z)\wedge!\,Before(x,z)\wedge!\,UseCup(a)\_(x,p,l)\\
DrinkTea\_(p,l,x,z)\wedge!\,Before(x,z)\wedge!\,UseBottle(b)\_(z,p,l)\\
DrinkTea\_(p,l,x,z)\wedge!\,Before(x,z)\wedge!\,Before(z,y)\\
\wedge!\,UseTeabag(c)\_(y,p,l)
\end{array} \tag{12}$$

5.3 Comparison of Gibbs sampling with MC-SAT algorithms

In order to infer the similar activities' accuracy, we have contrasted the two methods to find the applicable one. In the similar triggered activities' situation, the traditional MCMC–Gibbs sampling has a good performance in recognizing the two activities simultaneously. The comparison of Gibbs sampling and MCSAT of the two similar interleaved activities has been shown in Table 5. Table 6 is the result of the missing and disturbance situations' with Gibbs sampling and MC-SAT.

On the other hand, improving the accuracy of the missing data or disturbance situations, Gibbs has a better performance than MC-SAT. Therefore, for the complex activities, especially for similar activities and missing data or disturbance situations, the Gibbs sampling algorithm has a better performance. The Gibbs sampling algorithm has strong robustness which can be adopted in the future.

Table 4 The missing data or wrong data disturbance situation's inference precision of different activities

Activity	Preci. (no)	Preci. (with series and periods)
DrinkTea(~~*UseBottle*~~)	0.125	0.953
DrinkCoffee(*disturb*(*UseBottle*))	0.989	0.883
DoDishes(~~*DropWater*~~)	0.011	0.600
EatMeal(*disturb*(*UseFaucet*))	0.451	0.498
Sweep(~~*UseFaucet*~~)	0.125	0.995
Wipe(*disturb*(*UseMop*))	0.125	0.328

Table 5 The inference probability of the two similar interleaved activities

Activity	Preci. (Gibbs)	Preci. (MC-SAT)
DrinkTea&DrinkCoffee	0.438 & 0.466	0.961 & 0.00005
EatMeal&DoDishes	0.496 & 0.496	0.978 & 0.000049
Sweep&Wipe	0.700 & 0.245	0.973 & 0.000049

5.4 Discussion

We can find out from the above experiments that, as the two vital features of an activity, time series and time periods express different time dimensions by operations for the triggered time of entity events.

We can easily find the simplifying rules have extremely good performance of inference by the missing data of one activity, while it has the fluctuations in performance and even has the degradation in the disturbing situation. By comprehensive consideration, simplifying rules have better performance than the original rules.

We can find the Gibbs sampling algorithm has a better performance than MC-SAT. Because the two similar activities have many similar rules from entity events, always adopting the MC-SAT to handle the independence situations means the two interleaved activities are independent. That generates the conflicting results for the complex activities.

The better result relies on the rules and MLN features. Using the time, time series, periods, and location attributes to represent activity and the simplifying rules, the complex similar and data missing or disturbance situations' activities can be easily recognized. MLN is the typical hybrid method of the data-driven and knowledge-driven which draws the undirected graph structure that will make all rules more clearly and make the results more accurately.

Table 6 The missing data or wrong data disturbance situation's inference precision of different activities

Activity	Preci. (Gibbs)	Preci. (MC-SAT)
DrinkTea(~~*UseBottle*~~)	0.926	0.963
DrinkCoffee(*disturb*(*UseBottle*))	0.883	0.499
DoDishes(~~*DropWater*~~)	0.597	0.319
EatMeal(*disturb*(*UseFaucet*))	0.582	0.311
Sweep(~~*UseFaucet*~~)	0.996	0.995
Wipe(*disturb*(*UseMop*))	0.341	0.328

6 Conclusion

In this paper, the main work is realizing the complex activity recognition, including the similar activities, data missing, or disturbance situations by Markov logic network. In another important work, we have contrasted the typical methods, Gibbs sampling and MC-SAT algorithm, finding the best option in the complex situations. First of all, deploying the multi-modeling sensor in our daily life can collect many and varied data to improve the inference accuracy. For data processing, the hierarchical methodology is more efficient for MLN. The structure reduces many repetitive works. Especially for the computational of inference iteration, the exponential grow of the inference prefers the small size of data. In addition, MLN can establish the rule formulas for daily activity based on the time, time series, location, and period features of entity events. The simplifying rules have been adopted to improve the robustness and handle the uncertain situations. Then, MLN is the combination of the first-order logic and probability, construct Markov network, which has been given weight by the determinative learning method, including the Gibbs sampling and MC-SAT algorithms. Afterall, the complex activities have been recognized. We can find that the method has a typical advantage of the soften rules which means even though the personalization habit and data are missing all can be accepted and be solved.

This method is a typical hybrid inference which has the obvious improvement of the complex activities. For inhabitants, the MLN can reflect the preference and habits, except inferencing the activities; mining the relation between these activities is another important work. Also, the next work can go ahead to design the user portrait, especially for the multi-resident scenes. From another angle, recognizing the executor of one activity is a point which has a research value. We can do many extend work which will start and prepare based on this work.

Recognition of the daily activity is the fundamental work of many smart devices which is the basics and support for better service for human beings. We will continue to do the related work.

Abbreviations

ADL: Activity of daily living; DNF: Disjunctive normal form; FOL: First-order logic; MLN: Markov logic network; NP: Non-deterministic polynomial

Acknowledgements

Not applicable.

Authors' contributions

QL designed the solution, wrote this paper, and did the experimental tests. SC assisted the analysis of the algorithm. HN, TZ, and LC guided and checked the whole paper, experiments, and figures. All authors read and approved the final manuscript.

Funding

This work was supported by the National Natural Science Foundation of China (61872038 and 61811530335), and in part by the Fundamental Research Funds for the Central Universities under Grant FRF-BD-18-016A, and UK Royal Society-Newton Mobility Grant (No.IEC\NSFC\170067).

Availability of data and materials

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]School of Computer and Communication Engineering, University of Science and Technology Beijing, No.30, Xueyuan Road, Beijing, 100083 China. [2]China and Beijing Engineering Research Center for Cyberspace Data Analysis and Applications, No.30, Xueyuan Road, Beijing, 100083 China. [3]University of South China, No.28 Changsheng West Road, Hengyang, 421001 China. [4]Ulster University, Shore Road, Newtownabbey, BT37 0QB Northern Ireland.

Received: 2 July 2019 Accepted: 10 September 2019
Published online: 19 November 2019

References

1. S. Dhelim, H. Ning, M. A. Bouras, J. Ma, in *2018 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. Cyber-enabled human-centric smart home architecture, (2018), pp. 1880–1886. https://doi.org/10.1109/SmartWorld.2018.00316
2. E. Ruiz, V. Osmani, L. E. Sucar, O. Mayora, Detecting dressing failures using temporal–relational visual grammars. J. Ambient. Intell. Humanized Comput. **10**(7), 2757–2770 (2019). https://doi.org/10.1007/s12652-018-0975-0
3. L. Lu, H. Di, Y. Lu, L. Zhang, S. Wang, Spatio-temporal attention mechanisms based model for collective activity recognition. Signal. Process-Image Commun. **74**, 162–174 (2019). https://doi.org/10.1016/j.image.2019.02.01
4. A. Saeed, T. Ozcelebi, J. Lukkien, Synthesizing and reconstructing missing sensory modalities in behavioral context recognition. Sensors. **18**(9) (2018). https://doi.org/10.3390/s1809296
5. R. G. Hussain, M. A. Ghazanfar, M. A. Azam, U. Naeem, S. U. Rehman, A performance comparison of machine learning classification approaches for robust activity of daily living recognition. Artif. Intell. Rev. **52**(1), 357–379 (2019). https://doi.org/10.1007/s10462-018-9623-
6. D. Triboan, L. Chen, F. Chen, Z. Wang, A semantics-based approach to sensor data segmentation in real-time activity recognition. Futur. Gener. Comput. Syst. **93**, 224–236 (2019). https://doi.org/10.1016/j.future.2018.09.055
7. J. Ye, G. Stevenson, S. Dobson, KCAR: a knowledge-driven approach for concurrent activity recognition. Pervasive Mob. Comput. **19**, 47–70 (2015). https://doi.org/10.1016/j.pmcj.2014.02.003
8. D. Chakraborty, A. Zaslavsky, in *Dependable, Autonomic and Secure Computing, IEEE International Symposium on(DASC), vol. 00*. Saguna: Recognizing concurrent and interleaved activities in social interactions, (2011), pp. 230–237. https://doi.org/10.1109/DASC.2011.58. https://doi.ieeecomputersociety.org/10.1109/DASC.2011.58
9. D. Tian, X. Xu, Y. Tao, X. Wang, in *2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), vol. 1*. An improved activity recognition method based on smart watch data, (2017), pp. 756–759. https://doi.org/10.1109/CSE-EUC.2017.148
10. M. Babiker, O. O. Khalifa, K. K. Htike, A. Hassan, M. Zaharadeen, in *2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA)*. Automated daily human activity recognition for video surveillance using neural network, (2017), pp. 1–5. https://doi.org/10.1109/ICSIMA.2017.8312024
11. E. Kim, S. Helal, D. Cook, Human activity recognition and pattern discovery. IEEE Pervasive Comput. **9**(1), 48–53 (2010). https://doi.org/10.1109/MPRV.2010.7
12. T. Perumal, Y. L. Chui, M. A. B. Ahmadon, S. Yamaguchi, in *2017 IEEE 6th Global Conference on Consumer Electronics (GCCE)*. IoT based activity recognition among smart home residents, (2017), pp. 1–2. https://doi.org/10.1109/GCCE.2017.8229478
13. Z. Wang, S. Liu, J. Zhang, S. Chen, Q. Guan, A spatio temporal CRF for human interaction understanding. IEEE Trans. Circ. Syst. Video Technol. **27**(8), 1647–1660 (2017). https://doi.org/10.1109/TCSVT.2016.2539699

14. D. Sanchez, M. Tentori, J. Favela, Activity recognition for the smart hospital. IEEE Intell. Syst. **23**(2), 50–57 (2008). https://doi.org/10.1109/MIS.2008.18
15. M. Rodriguez, C. Orrite, C. Medrano, D. Makris, in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Fast simplex-HMM for one-shot learning activity recognition, (2017), pp. 1259–1266. https://doi.org/10.1109/CVPRW.2017.166
16. G. Okeyo, L. Chen, H. Wang, Combining ontological and temporal formalisms for composite activity modelling and recognition in smart homes. Futur. Gener. Comput. Syst. **39**, 29–43 (2014). https://doi.org/10.1016/j.future.2014.02.014. Special Issue on Ubiquitous Computing and Future Communication Systems
17. B. Bouchard, S. Giroux, A. Bouzouane, in *Advances in Artificial Intelligence*, ed. by L. Lamontagne, M. Marchand. A smart home agent for plan recognition (Springer, Berlin, 2006), pp. 25–36
18. C. Villalonga, R. M. Asif, A. K. Wajahat, H. Pomares, I. Rojas, S. Lee, B. L. Oresti, Ontology-based high-level context inference for human behavior identification. Sensors (Switserland). **16**(10), 1–26 (2016). https://doi.org/10.3390/s16101617
19. L. Chen, C. Nugent, G. Okeyo, An ontology-based hybrid approach to activity modeling for smart homes. IEEE Trans. Hum.-Mach. Syst. **44**(1), 92–105 (2014). https://doi.org/10.1109/THMS.2013.2293714
20. G. Civitarese, C. Bettini, T. Sztyler, D. Riboni, H. Stuckenschmidt, newnectar: collaborative active learning for knowledge-based probabilistic activity recognition. Pervasive Mob. Comput. **56** (2019). https://doi.org/10.1016/j.pmcj.2019.04.006
21. H.-S. Shin, D. Turchi, S. He, A. Tsourdos, Behavior monitoring using learning techniques and regular-expressions-based pattern matching. IEEE Trans. Intell. Transp. Syst. **20**(4), 1289–1302 (2019). https://doi.org/10.1109/TITS.2018.284926
22. T. Darpan, C. Liming, C. Feng, W. Zumin, Semantic segmentation of real-time sensor data stream for complex activity recognition. Pers. Ubiquit. Comput. **21**(3), 411–425 (2017). https://doi.org/10.1007/s00779-017-1005-5
23. V. Goranko, Logic in computer science: modelling and reasoning about systems. J. Logic Lang. Inf. **16**(1), 117–120 (2007)
24. B. Pfahringer, ed. by C. Sammut, G. I. Webb. Conjunctive Normal Form (Springer, Boston, 2017), pp. 260–261
25. P. Singla, P. Domingos, in *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference*. Discriminative Training of Markov Logic Networks[C] (AAAI Press, Pittsburgh, 2005), pp. 868–873
26. D. Lowd, P. Domingos, in *Knowledge discovery in databases: PKDD 2007*, ed. by J. N. Kok, J. Koronacki, R. Lopez de Mantaras, S. Matwin, D. Mladenič, and A. Skowron. Efficient weight learning for markov logic networks (Springer, Berlin, 2007), pp. 200–211
27. P. Domingos, S. Kok, D. Lowd, H. Poon, M. Richardson, P. Singla, ed. by L. De Raedt, P. Frasconi, K. Kersting, and S. Muggleton. Markov Logic (Springer, Berlin, 2008), pp. 92–117
28. L. Getoor, B. Taskar, *Markov Logic: a Unifying Framework for Statistical Relational Learning, vol. 0*. (MITP, Cambridge, 2007), pp. 339–371
29. S. D. Tran, L. S. Davis, in *Computer Vision – ECCV 2008*, ed. by D. Forsyth, P. Torr, and A. Zisserman. Event modeling and recognition using markov logic networks (Springer, Berlin, 2008), pp. 610–623
30. T. Geier, S. Biundo, in *2011 IEEE 23rd International Conference on Tools with Artificial Intelligence*. Approximate online inference for dynamic Markov logic networks, (2011), pp. 764–768. https://doi.org/10.1109/ICTAI.2011.120
31. H. Ning, F. Shi, T. Zhu, Q. Li, L. Chen, A novel ontology consistent with acknowledged standards in smart homes. Comput. Netw. **148**, 101–107 (2019). https://doi.org/10.1016/j.comnet.2018.11.004
32. D. Riboni, T. Sztyler, G. Civitarese, H. Stuckenschmidt, in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing. UbiComp '16*. Unsupervised recognition of interleaved activities of daily living through ontological and probabilistic reasoning (ACM, New York, 2016), pp. 1–12. https://doi.org/10.1145/2971648.2971691
33. K. S. Gayathri, S. Elias, R. Balaraman, Hierarchical activity recognition for dementia care using Markov logic network. Personal. Ubiquit. Comput. **19**(2), 271–285 (2015). https://doi.org/10.1007/s00779-014-0827-7

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.