

RESEARCH

A Hybrid Approach to Inferring the Internet of Things for Complex Activity Recognition

Qingjuan Li^{1,2}, Huansheng Ning^{1,2*}, Tao Zhu³, Shan Cui^{1,2} and Liming Chen⁴

*Correspondence:

ninghuansheng@ustb.edu.cn

¹School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing, China, No.30, Xueyuan Road, 100083 Beijing, China

Full list of author information is available at the end of the article

Abstract

With the rapid development and large-scale uptake of 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 is 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 modelling. Specifically, we develop a novel "Markov Logic Network" combining data-driven multi-feature and simplified rule-based modelling 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 activities groups, missing data or disturbance test datasets in a multi-modelling sensor scene. Initial results show our approach outperform 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 the 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 Ageing, 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-activity are more popular which require an active partition method for the continuous time sequence sensor data [6, 7, 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, 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.

Figure 1 A kitchen layout of multi-modeling sensors

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

Raw data collection is the first step, the second step is establishing the inference model. The usual inference methods adopted can be divided two categories, data-driven and knowledge-driven measures. Data-driven method trains the activity model which represent the implicit general rules by huge numbers of labeled data. Actually, the earliest studies have adopted the data-driven method mostly [13, 14, 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 includes Hidden Markov Models, 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 usually trained by the experimenters who is 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. These knowledge builds mapping from activity entity, appliance entity and context (sensor entity, sensor value entity), temporal-spatial traits, hierarchical methodology, rules (formulas) [16, 17]. This method has shown the good performance to use directly who overcomes "cold start" problem [18]. General rules of the model represent most residents' activities habit which have the high reusability. While, the preference, details, sufficient description of one activity cannot be taken into full consideration. For better performance, the activity model is dynamic and uncertainty which should be updated with the habits changing. How to match best results? Trying to combine the two method to learn from other's strong points to offset one's weakness which is called hybrid model [19]. Gabriele proposed a novel hybrid approach with probabilistic and knowledge-based reasoning, adopted the unsupervised feedback to label the rules which has

Figure 3 A water machine model of multi-modeling sensors

Figure 4 A cup model of multi-modeling sensors

the good performance in the real-world dataset [20]. Shin, Hyo-Sang presented the regular expressions-based string matching algorithm, the pattern model is trained by labeled data based on the experts knowledge base [21]. These hybrid methods are good performance for complement and correcting each other. Expert knowledge is more consistent which is utilized at the first step to model the basics formula [22]. Data modification is more flexible which revises the formula. In this paper, we adopt one hybrid model, Markov Logic Network (MLN), which use both of the first-order logic (FOL) and statistical probability. Establishing the rules' model by expert knowledge, then giving the satisfiability probability to every rule by learning from series of real raw sensor data sequences. Main advantages are the the strong ability for complex activities, having the good performance to handle the dynamic and the data missing. The details have 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, place, which covers all attributes of sensor data and reduces the decription complexity.
- In order to improve the robutness of activity model, we adopt the sub rules 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.
- Proving the Gibbs sampling algorithm has the better performance than traditional adopted method, MC-SAT algorithm in these complex activities.

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

2 Background (MLN)

2.1 Graphy Model of First-Order Logic

The first-order logic formula also calls as first order predicate calculus, consists of assertion and quantify. Constant symbols (e.g., time: 20181223, 20190107, etc.) represent the real instance. Variable symbols (e.g., x, y, etc.) are the abstraction of constant symbols which range over the real instance. Predicate symbols (e.g.m UseBootle, UseTeabag) represent the relations and attributes which mapping into a sets of category events, including sensor events, entity events, 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)

Figure 5 A deployed cup model of multi-modeling sensors

(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), 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, 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 who has the appear in at least one formula.

2.2 Learning Weight of 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 refine with less work. The learning algorithm is based on convex optimization. Gradient descent algorithm is one of the optimization method 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 the non-evidence atoms. For second-order class, the optimization method aims at searching direction from function as a quadratic surface [26].

$$\omega_{t+1} = \omega_t - \eta g \quad (1)$$

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

2.3 A Typical Method based First-Order Logic and Probability (MLN)

Markov Logic Network (MLN) is a combination of first-order logic and Markov Networks [27]. Its knowledge model is first-order formulas with weights. 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 $C = \{c_1, c_2, \dots, c_{|C|}\}$, 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, which means the probability is 0.

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

Reasoning probabilities for complex relationships is 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 adopts the second inference to calculate the every ground query atoms conditional probabilities. Gibbs sampling is a typical algorithm of Markov Chain Monte Carlo algorithm. MC-SAT algorithm is a slice sampling Markov Chain Monte Carlo algorithm which combining 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 i th ground formula when $X_l = x_l$ and $B_l = b_l$.

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

MLN has many outstanding advantages in followings:

- 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.

Figure 6 The flow chart of the novel method

Figure 7 Hierarchical Methodology for sensor, entity and activity

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, inference. The flow chart of the novel method has been shown in Figure 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 Figure 7.

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

$$Time, Place, SensorName, SensorID, Value \quad (5)$$

In order to recognize the similar activities, adding the time period feature and time series to the data. Combining the 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 sensor, place, time period has been shown in table 1, time including the year, month, day, hour, minute and second.

Table 1 Definition of the sensors, place and time period

Sensor	Place	Time Period
Motion	Bedroom	1,2,3,4
Touch	Livingroom	5,6,7,8
Light	Bathroom	9,10,11,12
Magnetic	Kitchen	0.5,1.5
Gas	Diningroom	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		

3.2 Simplifying the activity rules by entity expression

The typical entity event is action events which consists 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, Light**

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

- **Pressure**

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

- **Gas, Temperature, Humidity**

Enviornment change is key of the context awareness which is one kind of entity event, when the enviornment parameter cross the threshold value, the sensing data has been recorded which can provide the assistance for other change situations.

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

The entity event expression is shown in following:

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

3.3 Activity expression

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

The activity event expression is shown in following:

$$\begin{cases} Activity_{(Place, TimePeriod, BeginTime, EndTime)}, MeetsCondition \\ \neg Activity_{(Place, TimePeriod, BeginTime, EndTime)}, otherwise \end{cases} \quad (8)$$

3.4 Time series expression

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

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

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

The time relationship between the entity event (*entity_time*) and the activity event (*activity_begintime*, *activity_endtime*). We define the activity begintime and endtime has the relation as $Before(activity_begintime, activity_endtime)$. There has the five kinds which are incompatible between each other, Before, After, Between, Begin, End. The specific definition of them has shown in 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 rules expression (sub rules)

Reference [32] and [33] have adopted MLN to recognize the activities, but they works mainly for sequence activities without the probability learning which just a knowledge-driven method. Gabriele Civitarese has 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 rules, 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, noisy data. In order to improve the flexible of the activity model, the clausal form just includes the two event atoms which is a group sub rules 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 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 sub rules 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 evidenc entity event has been consisted into different rules which can has improved the flexibility of the inference.

4 Experiments

We designed 3 similar groups (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 living in the environment 2 weeks (as the small training data set to improve 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 bottom to top, connecting these sensors by Raspberry Pi, Arduino Mega 2560, Arduino Nano, communicating with PC by Serial Port Communication, Bluetooth HC-05, WIFI module, cellular network, storing the raw data in ".json" file. Segmenting the sensor data by the time-window (width = 10 seconds), then obtaining the entity event which has been stored into the ".db" file. Alchemy 2.0 is one of a engine to inference by MLN which we have used in this experiment.

Table 2 Activity-Entity-Sensor Design for 5 Activities

Activity	Entity	Sensor
Sweep	Usefaucet	rotate, velocity, water
	Use mop	touch
Wipe	Usefaucet	rotate, velocity, water
	Useduster	touch, pressure
Drink(Coffee, Tea)	Use cup	touch, tilt
	Useteabag	touch
	Use coffee bag	touch
	Use bottle	touch, pressure
Dishes	Bowl	touch, pressure
	Chopstick	touch, triaxial transducer
	Usefaucet	rotate, velocity, water
Eat	Bowl	touch, pressure
	Chopstick	touch, triaxial transducer

Our work is discussing the performance (average precision rate) of the time period for the 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, find the best method in the complex activity situations.

5 Results and Discussion

5.1 Similar activities by multi-attributes

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

//Evidence

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

//Query

$$\begin{aligned}
 & DrinkTea_{(place, period, begintime, endtime)} \quad (10) \\
 & DrinkCoffee_{(place, period, begintime, endtime)}
 \end{aligned}$$

//Rules

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

Definition 1 The term of the formula has two classes, variable uses all lowercase letters (e.g., x, y), constant using 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 first-order logic, the negation by \neg symbol, in MLN, that is same the ! symbol.

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

Table 3 Two similar activities probability

Activities Group	Preci.(no)	Preci.(with series and periods)
<i>DrinkTea</i> and <i>DrinkCoffee</i>	0.025	0.910
<i>HaveMeal</i> and <i>DoDishes</i>	0.011	0.992
<i>Sweep</i> and <i>Wipe</i>	0.025	0.975

5.2 Missing data and disturbance situations

In this part, simplifying the rules by dividing the rules to the common rules and special rules two part. Because of the vital role of the special rules, initial weight value is bigger than common rules, the connection weight value between common and special rules is same with the common rules. We have tested the missing data for different activities, the accuracy has a significant improvement which has been in Table 4.

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

//Simplifying Rules

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

(12)

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

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

5.3 Comprison the Gibbs sampling with MC-SAT algorithms

In order to infer the similar activities accuracily, we have contrasted the two methods to find the applicable one. In the similar triggered activities situation, the traditional MCMC–Gibbs sampling has the good performance to recognize the two activities simultaneously.

Table 5 The inference probability of the two similar interleaved activities

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

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

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

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

5.4 Discussion

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

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

We can find the Gibbs sampling algorithm has the better performance than MC-SAT. Because the two similar activities has the many similar rules from entity events, always adopting the MC-SAT to handling the independence situations which 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, location attributes to represent activity and the simplifying rules, the

complex similar and data missing or disturbance situations' activities can be easily recognition. 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.

6 Conclusion

In this paper, the main work is realizing the complex activities 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 which 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, period features of entity events. The simplifying rules has 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 method has a typical advantage of the soften rules which means even though the personalization habit and data-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 inferecing 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 the research value. We can do many extend work which will start and prepare based 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

FOL: First-order logic

MLN: Markov Logic Network

DNF: Disjunctive normal form

NP: Non-deterministic polynomial

Declarations

Availability of data and materials

Not applicable.

Competing interests

No conflict of interest exists in the submission of this manuscript, and manuscript is approved by all authors for publication.

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).

Author's contributions

Qingjuan Li designed the solution, wrote this paper, did the experimental tests. Shan Cui assisted the analysis of algorithm. Huansheng Ning, Tao Zhu, Liming Chen guided and checked the whole paper, experiments and figures. All authors read and approved the final manuscript.

Acknowledgements

Not applicable.

Author details

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

References

- Dhelim, S., Ning, H., Bouras, M.A., Ma, J.: Cyber-enabled human-centric smart home architecture. 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), pp. 1880–1886 (2018). doi:10.1109/SmartWorld.2018.00316
- Ruiz, E., Osmani, V., Sucar, L.E., Mayora, O.: Detecting dressing failures using temporal–relational visual grammars. *Journal of Ambient Intelligence and Humanized Computing* 10(7), 2757–2770 (2019). doi:10.1007/s12652-018-0975-0
- Lu, L., Di, H., Lu, Y., Zhang, L., Wang, S.: Spatio-temporal attention mechanisms based model for collective activity recognition. *SIGNAL PROCESSING-IMAGE COMMUNICATION* 74, 162–174 (2019). doi:10.1016/j.image.2019.02.01
- Saeed, A., Ozcelebi, T., Lukkien, J.: Synthesizing and Reconstructing Missing Sensory Modalities in Behavioral Context Recognition. *SENSORS* 18(9) (2018). doi:10.3390/s1809296
- Hussain, R.G., Ghazanfar, M.A., Azam, M.A., Naeem, U., Rehman, S.U.: A performance comparison of machine learning classification approaches for robust activity of daily living recognition. *ARTIFICIAL INTELLIGENCE REVIEW* 52(1), 357–379 (2019). doi:10.1007/s10462-018-9623-3
- Triboan, D., Chen, L., Chen, F., Wang, Z.: A semantics-based approach to sensor data segmentation in real-time activity recognition. *Future Generation Computer Systems* 93, 224–236 (2019). doi:10.1016/j.future.2018.09.055
- Ye, J., Stevenson, G., Dobson, S.: Kcar: A knowledge-driven approach for concurrent activity recognition. *Pervasive and Mobile Computing* 19, 47–70 (2015). doi:10.1016/j.pmcj.2014.02.003
- Chakraborty, D., Zaslavsky, A., Saguna: Recognizing concurrent and interleaved activities in social interactions. In: Dependable, Autonomic and Secure Computing, IEEE International Symposium on(DASC), vol. 00, pp. 230–237 (2011). doi:10.1109/DASC.2011.58. doi.ieeecomputersociety.org/10.1109/DASC.2011.58
- Tian, D., Xu, X., Tao, Y., Wang, X.: An improved activity recognition method based on smart watch data. In: 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), vol. 1, pp. 756–759 (2017). doi:10.1109/CSE-EUC.2017.148
- Babiker, M., Khalifa, O.O., Htike, K.K., Hassan, A., Zaharadeen, M.: Automated daily human activity recognition for video surveillance using neural network. In: 2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA), pp. 1–5 (2017). doi:10.1109/ICSIMA.2017.8312024
- Kim, E., Helal, S., Cook, D.: Human activity recognition and pattern discovery. *IEEE Pervasive Computing* 9(1), 48–53 (2010). doi:10.1109/MPRV.2010.7
- Perumal, T., Chui, Y.L., Ahmadon, M.A.B., Yamaguchi, S.: lot based activity recognition among smart home residents. In: 2017 IEEE 6th Global Conference on Consumer Electronics (GCCE), pp. 1–2 (2017). doi:10.1109/GCCE.2017.8229478
- Wang, Z., Liu, S., Zhang, J., Chen, S., Guan, Q.: A spatio temporal crf for human interaction understanding. *IEEE Transactions on Circuits and Systems for Video Technology* 27(8), 1647–1660 (2017). doi:10.1109/TCSVT.2016.2539699
- Sanchez, D., Tentori, M., Favela, J.: Activity recognition for the smart hospital. *IEEE Intelligent Systems* 23(2), 50–57 (2008). doi:10.1109/MIS.2008.18
- Rodriguez, M., Orrite, C., Medrano, C., Makris, D.: Fast simplex-hmm for one-shot learning activity recognition. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1259–1266 (2017). doi:10.1109/CVPRW.2017.166
- Okeyo, G., Chen, L., Wang, H.: Combining ontological and temporal formalisms for composite activity modelling and recognition in smart homes. *Future Generation Computer Systems* 39, 29–43 (2014). doi:10.1016/j.future.2014.02.014. Special Issue on Ubiquitous Computing and Future Communication Systems

17. Bouchard, B., Giroux, S., Bouzouane, A.: A smart home agent for plan recognition. In: Lamontagne, L., Marchand, M. (eds.) *Advances in Artificial Intelligence*, pp. 25–36. Springer, Berlin, Heidelberg (2006)
18. Villalonga, C., Asif, R.M., Wajahat, A.K., Pomares, H., Rojas, I., Lee, S., Oresti, B.L.: Ontology-based high-level context inference for human behavior identification. *Sensors (Switzerland)* 16(10), 1–26 (2016). doi:10.3390/s16101617. 10.3390/s16101617
19. Chen, L., Nugent, C., Okeyo, G.: An ontology-based hybrid approach to activity modeling for smart homes. *IEEE Transactions on Human-Machine Systems* 44(1), 92–105 (2014). doi:10.1109/THMS.2013.2293714
20. Civitaresse, G., Bettini, C., Szt Tyler, T., Riboni, D., Stuckenschmidt, H.: newnectar: Collaborative active learning for knowledge-based probabilistic activity recognition. *Pervasive and Mobile Computing* 56 (2019). doi:10.1016/j.pmcj.2019.04.006
21. Shin, H.-S., Turchi, D., He, S., Tsourdos, A.: Behavior Monitoring Using Learning Techniques and Regular-Expressions-Based Pattern Matching. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 20(4), 1289–1302 (2019). doi:10.1109/TITS.2018.284926
22. Darpan, T., Liming, C., Feng, C., Zumin, W.: Semantic segmentation of real-time sensor data stream for complex activity recognition. *Personal and Ubiquitous Computing* 21(3), 411–425 (2017). doi:10.1007/s00779-017-1005-5
23. Goranko, V.: Logic in computer science: Modelling and reasoning about systems. *Journal of Logic, Language and Information* 16(1), 117–120 (2007)
24. Pfahringer, B.: In: Sammut, C., Webb, G.I. (eds.) *Conjunctive Normal Form*, pp. 260–261. Springer, Boston, MA (2017)
25. Singla, P., Domingos, P.: Discriminative training of markov logic networks., vol. 2, pp. 868–873 (2005)
26. Lowd, D., Domingos, P.: Efficient weight learning for markov logic networks. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenić, D., Skowron, A. (eds.) *Knowledge Discovery in Databases: PKDD 2007*, pp. 200–211. Springer, Berlin, Heidelberg (2007)
27. Domingos, P., Kok, S., Lowd, D., Poon, H., Richardson, M., Singla, P.: In: De Raedt, L., Frasconi, P., Kersting, K., Muggleton, S. (eds.) *Markov Logic*, pp. 92–117. Springer, Berlin, Heidelberg (2008)
28. Getoor, L., Taskar, B.: *Markov Logic: A Unifying Framework for Statistical Relational Learning*, vol. 0, pp. 339–371. MITP, Cambridge, MA (2007)
29. Tran, S.D., Davis, L.S.: Event modeling and recognition using markov logic networks. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *Computer Vision – ECCV 2008*, pp. 610–623. Springer, Berlin, Heidelberg (2008)
30. Geier, T., Biundo, S.: Approximate online inference for dynamic markov logic networks. In: *2011 IEEE 23rd International Conference on Tools with Artificial Intelligence*, pp. 764–768 (2011). doi:10.1109/ICTAI.2011.120
31. Ning, H., Shi, F., Zhu, T., Li, Q., Chen, L.: A novel ontology consistent with acknowledged standards in smart homes. *Computer Networks* 148, 101–107 (2019). doi:10.1016/j.comnet.2018.11.004
32. Riboni, D., Szt Tyler, T., Civitaresse, G., Stuckenschmidt, H.: Unsupervised recognition of interleaved activities of daily living through ontological and probabilistic reasoning. In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing. UbiComp '16*, pp. 1–12. ACM, New York, NY, USA (2016). doi:10.1145/2971648.2971691
33. S., G.K., Elias, S., Balaraman, R.: Hierarchical activity recognition for dementia care using markov logic network. *Personal and Ubiquitous Computing* 19(2), 271–285 (2015). doi:10.1007/s00779-014-0827-7