Using model's temporal features and hierarchical structure for similar activity recognition

Qingjuan Li[1,2] · Huansheng Ning[1,2] · Lingfeng Mao[1] · Liming Chen[3]

Abstract
Using Sensor-based approach in activity recognition usually requires the deployment of many ambient sensors to objects and environments. Each sensor can be triggered by more than one activity, e.g., a touch sensor of a cooker can be triggered by cooking, doing dish and so on. An activity consists of some sensor events. When the number of same sensors are in the majority of two activities, the two activities are defined as similar activities which are difficult to distinguish. To address the challenge of recognizing similar activities, this paper conceives a new activity recognition approach incorporating high-dimensional features of duration and time block characteristics to improve the inference performance. In a further step, we take advantage of these similar activities to build a hierarchical structure model which can improve capacities of expandability and standardization. We design experiments of similar activity in our daily life to evaluate this solution. The results show that high-dimensional temporal features improved similar activity inference accuracy on an average of 1.88 times, and the use of hierarchical structure can generalize specific rules to standard ones which decreases similar activity recognition computation time on an average of 0.36 times.

✉ Huansheng Ning
ninghuansheng@ustb.edu.cn

Qingjuan Li
ustbliqingjuan@sina.com

Lingfeng Mao
lingfengmao@ustb.edu.cn

Liming Chen
l.chen@ulster.ac.uk

1 University of Science and Technology Beijing, Beijing, China

2 Beijing Engineering Research Center for Cyberspace Data Analysis and Applications, Beijing, China

3 Ulster University, Newtownabbey, UK

1 Introduction

With the development of the Internet of Things, sensor technology is being widely used in our daily life (Suryadevara and Mukhopadhyay 2014; Pansiot et al. 2007; Gao et al. 2014). Data mining, information inference and knowledge learning have been risen in response to the proper time and conditions of the smart world. More and more researches adopt the non-vision ambient sensors in their family scene which pays attention to resident privacy. However, there are many limitations in recognizing complex activities, such as, the noisy interference situation, the indistinguishable similar activity. An activity consists of a series of sensor elements. Similar activities are two activities which contain more than half same sensor elements in their sensor series. A small number of sensor elements are different in similar activities. The same sensor elements mean the sensor belongs to these two activities. The touch sensor of cup is a same sensor element of several activities in same activity category (drink coffee and drink tea), or in different activity categories (brush teeth and drink coffee).

Most of previous studies focus on some typical activities with poor correlations in single resident and sequence

scenario. They usually adopt CASAS dataset which includes ten independent and not similar activities, like making meal and eating meal (Cook et al. 2013). There are some other datasets with similar activities have been adopted. For example, Liming Chen et al. have designed their own dataset in kitchen room with eight activities. Three activities of them, make tea, make chocolate and make coffee, are similar activities. But they have not designed the experiments specifically for these similar activities (Okeyo et al. 2014a). Two similar activities occurred in one time window is a usual situation which needs a particular dataset. Although similar activity recognition is at an initial stage, but its essence is mining features of data to establish the high correlations with the right activity which is same with the traditional activity recognition.

In addition to sensor, location (Tahir et al. 2019) and other intuitive features of sensor data, time sequence (Moutacalli et al. 2015) is an effective feature which has been studied for concurrent and interleaved activities by Saguna (2013), Yongmian Zhang (2013), Li Liu (2016) and others. However, once a sensor is triggered in one window, the related similar activity will have the same probabilities (calculate by the maximum conditional probability from the labeled dataset). Similar activities always have a number of same time sequence orders, sensors and locations which makes the distinguishing between them represents a difficult task.

The new feature, time duration, has been proposed. Fadi Al Machot et al. have adopted the Information Gain (IG) evaluation to find the set of "best fitting sensors" (Machot et al. 2016). Li Junhuai et al. have divided the activities to basic and transitional activities, after running the shortest segmenting on the raw sensor data, using the K-Means cluster analysis to gather the related segments as the basic activities' time blocks (Junhuai et al. 2019). These two methods calculate all the possible results to get the best which is complex and high computation. Surong Yan et al. have combined the Latent Dirichlet allocation and Bayes theorem to represent and extract activity duration feature (Yan et al. 2016). However, these duration features are fixed values which can't handle the dynamic situations. Ehsan has proposed a normal distribution model of temporal features and activities, like, time sequences, begin times and durations based on their sensor data. However, this method has poor generalization ability which requires a knowledge model (Nazerfard 2018). Defining alternative and duration range model can reduce reliance on data, and increase the flexibility. In the high-dimensional time features, time block has been introduced to express the duration range is an innovation point in our model.

After feature selecting, algorithm choosing is also a key process for accurate activity recognition. There are two categories of algorithms, one is data-driven method, and the other one is knowledge-based method. The semantic model with the temporal-spatial and time sequence traits is a typical knowledge-based method which design the activity rules in advance and not rely on user data (Liu et al. 2015). Hooda et al. have proposed the ontology model to express the heterogeneous sensor data which has reusability (Hooda and Rani 2020). Using the probability statistics is the basic idea of data-driven method which has a good performance in dynamic and unknown case (Chamroukhi et al. 2013). The combination of semantic and probability statistics algorithm is the promising method of the inference, especially for the complex representation and relationship of activities situation (Okeyo et al. 2014b; Riboni and Bettini 2011; Ordóñez et al. 2013; Meditskos et al. 2013). Markov Logic Network (MLN) is a combination solution which has been widely adopted (Gayathri et al. 2017; Helaoui et al. 2011). These studies are mainly handling the activity recognition in interleaved and concurrent scenes. In order to better understand how it can be applied to similar activity recognition, Markov Logic Network has been elaborated in Sect. 2.

With the deepening of the research, more and more detailed activities are involved in the model whose scale increases greatly with the redundant representation of similar activities (Chen and Nugent 2009). The related rules of an activity consist of special habit rules and the complete homologous rules for same category activities. That reduces the consumption of the resource and the complexity of these rules which builds the formal management for these activities (Ye et al. 2015). It can be found that generalization for these similar activities generates homologous rules which have better representation than semantic rules (depend more on expert knowledge than data) for dynamically unknown activities.

In this paper, we improve Markov Logic Network model as described in the following steps.

1. Adding temporal characteristics, such as duration and time block of an activity, to activity models. This trait can increase the correlations between sensor and activity which can distinguish the similar activity easily.
2. Proposing a novel hierarchical structure and improving the model robustness and generalization.

The basic concept and theory of MLN algorithm has been presented in Sect. 2. In Sect. 3, the semantic activity representation has been presented including the time duration and time block. The hierarchical structure based same category rules and special derivative rules is explained in Sect. 4. Section 5 shows the experiment results for similar activity based on the Markov Logic Network model which has good performance. In Sect. 6, we discuss the solution and propose directions for future work.

2 Markov Logic Network

In this section, the basic concepts of MLN have been described, including the knowledge representation method and probabilistic reasoning logic. Knowledge representation is the fundamental for characteristics and hierarchical structure. Probabilistic reasoning logic is the key of accurate inference.

Markov Logic Network is one kind of Markov Network (MN) whose rules are expressed by First order Logic (FoL) (Tran and Davis 2008; Chahuara et al. 2012; Gayathri et al. 2015).

First order Logic is a knowledge representation model which is built by connector (e.g, $\wedge, \vee, \neg, \rightarrow, \leftrightarrow$) and quantifiers (e.g, $\forall, \exists$) recursively. The complete representation contains types of terms, for example, constant, variable, function, etc. Variable is the generalization of constants which has the same correlations or attributes. The function represents mappings from tuples of objects to objects (Domingos and Lowd 2009). Predicate expresses the correlation and attributes of terms (Domingos et al. 2008). Each term represents a node of MLN, each predicate represents a edge of MLN which link all the terms in one FoL rule. An MLN is an undirected graph. Each FoL rule represents a fully-connected graph called "clique". The ground term is a constant term without any variables.

We construct the MN based the FoL formula and then give the weight (related to the potential function) for every formula which represents the occurrence probability of them based the label data. Weight ω has the following relationship with potential function $\Phi_k(x_{\{k\}})$. Therefore, MLN also defined as the combination of FoL and a set of potential functions. The potential functions represent the relational degree for the linked nodes which is non-negative real-valued function of the state. The potential function is applied to pairwise nodes in one FoL

$$\omega = log\Phi_k(x_{\{k\}}) \tag{1}$$

There are two kinds of methods to obtain the weight of MLN, one is manually set, and the other one learns by learning algorithms automatically. We adopt the second one which can obtain much better models with less work (Domingos and Lowd 2009). We adopt the discriminative weight learning method where some atoms are evidence, and the others are queried to achieve our goal in predicting the latter from the former. The MN usually represents as log-linear probability models. Maximizing the conditional log-likelihood is an optimization method for learning weight. The weight "ω" has the following formula with the learning rate "η" and gradient "g" (Singla and Domingos 2005)

$$\omega_{t+1} = \omega_t - \eta g \tag{2}$$

The gradient "g" is obtained by taking the derivative for the conditional probability of the unknown atoms y and known evidence x. g is the difference of the expected number of true groundings of the corresponding clause $\sum_{y'} P_\omega(Y = y' \mid X = x)n_i(x, y')$ and the actual number $n_i(x, y)$. $E_{\omega,y}$ is the expectation over the non-evidence atoms Y. $n_i(x, y)$ is the number of true groundings of the ith formula in the data.

$$
\begin{aligned}
g &= \frac{\partial}{\partial \omega_i}(-logP_\omega(Y = y \mid X = x)) \\
&= -n_i(x, y) + \sum_{y'} P_\omega(Y = y' \mid X = x)n_i(x, y') \\
&= E_{\omega,y}[n_i(x, y)] - n_i(x, y)
\end{aligned}
\tag{3}
$$

Inference in MLN is a non-deterministic polynomial hard (NP-hard) problem which requires the sampling method. Gibbs sampling is the typical method that we adopt in this paper. Gibbs sampler ensures the conditioning variables fixing to their given values. The details of this algorithm are shown in following. The sample sequence is approximated by iterative conditional distribution and joint distribution.

Algorithm 1 Gibbs Sampling

1: **function** SAMPLING(F, L, C) ► Where F - rule, L - Logic of MLN, C - Constant of MLN

2:       Random initialization $X_0 = x_0, Y_0 = y_0$

3:       **while** $t = 0,1,2,...$ **do**

4:             $y_{t+1} \sim p(y|x_t)$

5:             $x_{t+1} \sim p(x|y_{t+1})$

6:       **end while**

7: **end function**

In order to reduce the computing scale, sampling in Markov blanket is an efficient method for inference. Markov blanket is the minimal set of nodes that renders one specific node independent of the remaining network. The probability of a ground predicated (query nodes) X_l when its Markov blanket (related evidence nodes which has smaller number than MLN evidence nodes) B_l is in state b_l is in (4). F_l is the set of ground formulas that X_l appears in, ω_i is the weight of clique of one formula, and $f_i \in \{0, 1\}$ is a binary function which represents the state of clique. $f_i(X_l = x_l, B_l = b_l)$ is the value of ith ground formula when $X_l = x_l$ and $B_l = b_l$. $f_i(X_l = 0, B_l = b_l)$ is the value of ith ground formula when $X_l = 0$ and $B_l = b_l$. $f_i(X_l = 1, B_l = b_l)$ is the value of ith ground formula when $X_l = 1$ and $B_l = b_l$

$$P(X_l = x_l | B_l = b_l) = \frac{exp\left(\sum_{f_i \in F_l} \omega_i f_i(X_l = x_l, B_l = b_l)\right)}{exp\left(\sum_{f_i \in F_l} \omega_i f_i(X_l = 0, B_l = b_l)\right) * exp\left(\sum_{f_i \in F_l} \omega_i f_i(X_l = 1, B_l = b_l)\right)} \quad (4)$$

3 Semantic model with duration and time block

Semantic representations for all activities adopt FoL format in MLN (Ryoo and Aggarwal 2009; Gayathri et al. 2017). In the sensor event layer, the sensor attributes (time point, location, time block, ID, attached object) have been defined as parts of term. The time sequence has been defined as a new term which can be linked by predicates with sensor term or activity term. We only recorded the jump value of these terms and discarded the ones that do not change, which saves storage and computation resources. Most of sensors have two states, we redefine these sensor terms' states, 1 means from untriggered to triggered, 0 means from triggered to untriggered. While, pressure sensor, temperature sensor and other similar sensors have values instead of states, we transform these values to term states, 1 means the value has been increased, 0 means the value has been decreased. The two state terms are expressed in function (5–6). These two functions are opposites of each other which

Table 1 Time block definition

Time block	Time slot (o'clock)	Time block	Time slot (o'clock)
1	0–2	2	2–4
3	4–6	4	6–8
5	8–10	6	10–12
7	12–14	8	14–16
9	16–18	10	18–20
11	20–22	12	22–24

has been shown in function (7). There are more than ten sensor categories, such as motion, touch, light, magnetic, gas, water, pressure, tilt, temperature, humidity and vibration. DD, PP, $HHMMSS$ is the format of the time information used to show the time information, including the traditional temporal data DD, $HHMMSS$, day, hour, minute and second. The new concept of the time block PP which has the twelve value (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12) is divided by 2 h in 1 day as shown in Table 1

$$Sensor(ID)Place(LABEL)(DD, PP, HHMMSS) \quad (5)$$

$$\neg Sensor(ID)Place(LABEL)\_ID(DD, PP, HHMMSS) \quad (6)$$

$$Sensor(ID)Place(LABEL)(DD, PP, HHMMSS) \leftrightarrow \neg(\neg Sensor(ID)Place(LABEL)(DD, PP, HHMMSS)) \quad (7)$$

All terms, which are constant, are named ground atom. For example, when the cup has detected a touch from touch sensor 1, the cup's touch sensor changed the value from "0" to "1", then it sends one record. The date is 20190212, the time block is 2 (Morning), the hour is 09, the minute is 15, and the second is 12 as shown in function (8)

$$Touch(1)Cup(1)(20190212, 2, 091512) \quad (8)$$

In entity/action event layer, the atom is similar to the sensor event layer and shown in function (9). For example, the action "UsingCup" is shown in function (10) which has been used to infer. We define the action time is the last triggered sensor time

$$Action(ID)Place(LABEL)(DD, PP, HHMMSS) \quad (9)$$

$$Touch(1)Cup(1)(x, y, z_1) \wedge \neg Megnetic(1)Coster(1)(x, y, z_2)$$
$$\wedge z_1 \leq z_2 \rightarrow Using(1)Cup(1)(x, y, z_2) \quad (10)$$

Additional Notes One action only occurs at one time point, but it can belong to three time blocks.

For example, 06:32:45 (HH:MM:SS) is one time point for one action, this action, belongs to 4, 3.5, and 4.5 three time blocks (3.5 and 4.5 will be give the definition in following). Therefore, the preprocessing for the raw data extends them to three instances. This method aims at solving the issue of the representation for crossing time blocks activity, even though that will waste some

Table 2 New time block definition

New time block	Cross border time block	Time block	Cross border time block
0.5	0–1	1.5	1–2
2.5	2–3	3.5	3–4
4.5	4–5	5.5	5–6
6.5	6–7	7.5	7–8
8.5	8–9	9.5	10–11
10.5	10–11	11.5	11–12

storage resources. The efficiency of computation has been improved sharply because of the unification rules.

The activity event layer, is the same as entity and sensor layer, except adding the duration concept to the atom (Duong et al. 2006, 2009; Zhang et al. 2010). The activity is defined as (11), *Begin_HHMMSS* and *End_HHMMSS* have been provided from series of action events. The knowledge rules of the activities consist of several action events. Selecting the minimum time and the maximal time is set as the activity's begin time and end time. Usually, one activity consists of more than one action event, therefore, the max and min time must exist. For the exception that only has one action event, we define the begin and end time the same is

the event time. In order to reduce the character numbers and lower the limit, we combine the duration and the time block, which adds the 12 new values shown in Table 2. The duration is a loose time frame which lasting less than 4 h, when one activity happens in cross time block, they are exists in the new time block. The definition for crossing duration has been shown in following.

$$Activity(ID)Place(LABEL)$$
$$(DD, PP, Begin\_HHMMSS, End\_HHMMSS) \tag{11}$$

The typical activity rules of 12 activities (*DrinkTea*, *DrinkCoffee*, *WashFace*, *WashCloth*, *HaveMeal*, *DoDishes*, *DrinkMilk*, *DrinkJuice*, *FriedDishes*, *BoiledDishes*, *Sweep* and *Wipe*) that are used in experiments have been shown in Table 3. In order to make the rules clearer, we just keep the entity name, time sequence and activity name.

4 Hierarchical activity modelling

The structuring of the activity model aims at establishing the abstract and generalized rules based on classifying the categories of the similar Activities of Daily Living (ADL) as shown in Fig. 1 (Brostow et al. 2008). For example, we defined a

Table 3 Typical rules of 12 activities

| Typical rules |
|---|
| $UseTeabag(T1) \wedge UseBottle(T2) \wedge PourBottle(T3) \wedge UseCup(T4) \wedge DecreaseCup(T5) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \wedge Before(T3, T4) \wedge Before(T4, T5) \rightarrow \mathbf{DrinkTea}(T1, T5)$ |
| $UseCoffeebag(T1) \wedge UseBottle(T2) \wedge PourBottle(T3) \wedge UseCup(T4) \wedge DecreaseCup(T5) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \wedge Before(T3, T4) \wedge Before(T4, T5) \rightarrow \mathbf{DrinkCoffee}(T1, T5)$ |
| $TwistFaucet(T1) \wedge UsePool(T2) \wedge UseCleanser(T3) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \rightarrow \mathbf{WashFace}(T1, T3)$ |
| $TwistFaucet(T1) \wedge UsePool(T2) \wedge UseDetergent(T3) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \rightarrow \mathbf{WashCloth}(T1, T3)$ |
| $UseChair(T1) \wedge UseDiningtable(T2) \wedge UseChopstick(T3) \wedge UseBowl(T4) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \wedge Before(T3, T4) \rightarrow \mathbf{HaveMeal}(T1, T4)$ |
| $UsePot(T1) \wedge UseChopstick(T2) \wedge UseBowl(T3) \wedge TwistFaucet(T4) \wedge UsePool(T5) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \wedge Before(T3, T4) \wedge Before(T4, T5) \rightarrow \mathbf{DoDishes}(T1, T5)$ |
| $UseMilkbottle(T1) \wedge PourMilkbottle(T2) \wedge UseCup(T3) \wedge DecreaseCup(T4) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \wedge Before(T3, T4) \rightarrow \mathbf{DrinkMilk}(T1, T4)$ |
| $UseJuicebottle(T1) \wedge PourJuicebottle(T2) \wedge UseCup(T3) \wedge DecreaseCup(T4) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \wedge Before(T3, T4) \rightarrow \mathbf{DrinkJuice}(T1, T4)$ |
| $UsePot(T1) \wedge UseOil(T2) \wedge MovePot(T3) \wedge UseCO(T4) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \wedge Before(T3, T4) \rightarrow \mathbf{FriedDishes}(T1, T4)$ |
| $UsePot(T1) \wedge TwistFaucet(T2) \wedge MovePot(T3) \wedge UseCO(T4) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \wedge Before(T3, T4) \rightarrow \mathbf{BoiledDishes}(T1, T4)$ |
| $UseBroom(T1) \wedge MoveBroom(T2) \wedge Before(T1, T2) \rightarrow \mathbf{Sweep}(T1, T2)$ |
| $UseMop(T1) \wedge TwistFaucet(T2) \wedge UsePool(T3) \wedge$ |
| $Before(T1, T2) \wedge Before(T2, T3) \rightarrow \mathbf{Wipe}(T1, T3)$ |
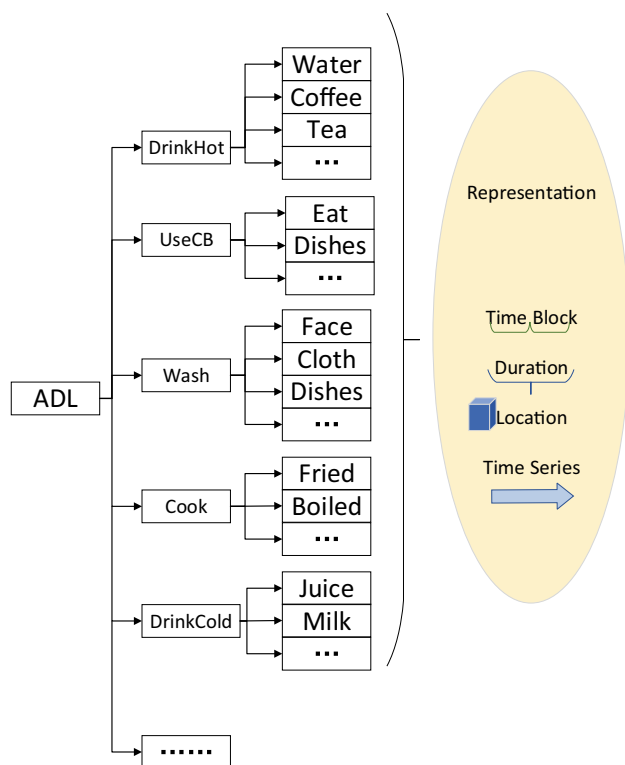
Fig. 1 Activity categories

category activity "DrinkHot" based on similar activities which includes Activity (drink water), Activity (drink tea), Activity (drink coffee), and so on, the generalized rule of "DrinkHot" as shown in Table 4. The "DrinkTea" rule has been redefined by "DrinkHot" which has been shown in Table 5. We can find the representation of the special rules is easier than before. In this paper, we just list some typical categories which may be incomplete, but has the same processing method and can be extended in all ADL.

We can see that each type of ADL consists of many detailed and specific activities. For one kind, there are many sub activities which triggers different sensors that not only obey the generalized rules, but also meet the special rules. According to

living habits, common sense of ADL, the semantic knowledge of ADL can be easily established and enriched.

For the father nodes, extracting the generalized rules from the sub nodes, adopts the FoL to describe them. For the leaf nodes, the complex description can be replaced by the father nodes and personal characters with connectives which have been shown in function (12). There are four typical features for every activity node, Time Block, Duration, Location and Time Series.

Adopting the hierarchical structure model has many advantages as following.

- **Ease of maintenance**, sub activities inherit rule models from father node which doesn't influence the special feature of sub. When activity habits change, model maintenance is convenient and low-cost because of readability and inheritance.
- **High expandability**, when the father category is defined, the father node is expended to various sub nodes. The model is flexible and low-cost to realize the high cohesion relationship with father and sub nodes.
- **High reusability**, the father node is independent and can be reused by new activities that have the same features.
- **High efficiency**, because of the high expandability and reusability, the whole operation time has been reduced.
- **Multiple inheritance**, each specific activity not only inherit one father node attributes, but also can belong to more father nodes which means they can get all of their fathers' attributes. That enhances the robustness of the model.

$$Father\_Activity \wedge Special\_Character \rightarrow Sub\_Activity \quad (12)$$

5 Inference and experiment

In this section, we develop an inference method which combines the data and knowledge reasoning. FoL has been presented as a typical semantic model. We establish the expert's

Table 4 Typical DrinkHot category activity rule

DrinkHot typical rule
$UseBottle(T1) \wedge PourBottle(T2) \wedge UseCup(T3) \wedge DecreaseCup(T4) \wedge$ $Before(T1, T2) \wedge Before(T2, T3) \wedge Before(T3, T4) \rightarrow \mathbf{DrinkHot}(T1, T4)$

Table 5 Typical DrinkTea activity rule based on DrinkHot category

DrinkTea typical rule
$UseTeabag(T1) \wedge DrinkHot(T2) \wedge Before(T1, T2) \rightarrow \mathbf{DrinkTea}(T1, T2)$

knowledge base based on the essence and nature characteristics of these activities which is completely unaffected by the sensor data. In order to decrease the space usage and computation complexity, a hierarchical structure activity model has been presented to modifying the rules of knowledge base. The activity consists of a series of sensor data, and the most critical feature is time series. Therefore, MN is a statistical probability graphic method has been adopted in this paper. It can dig the complex and personal features from the sensor data. By combining the FoL and MN, we have adopted the MLN which has a good performance in recognizing complex activities.

Our experiment has two research points, one is recognizing the similar activity which happen in one day, and the other one is hierarchical structure model performance improvement. We deploy 27 sensors in our room, including touch sensor (TTP223B), tilt sensor, magnetic (MKA14103), water (FC-37), pressure (HX711) and so on. The simulation deployment diagram of the room is shown in Fig. 2. These sensors are divided into the module boxes and deployed in their families. Similar activity groups consist of Activity (*DrinkTea*) and Activity (*DrinkCoffee*), Activity (*WashFace*) and Activity (*WashCloth*), Activity (*HaveMeal*) and Activity (*DoDishes*), Activity (*DrinkMilk*) and Activity (*DrinkJuice*), Activity (*FriedDishes*) and Activity (*BoiledDishes*), Activity (*Sweep*) and Activity (*Wipe*) which have similar actions more than different ones. We extract the similar parts to be used as the father features, whereas the remaining parts are used as particular characters of sub activities.

Because of the difference between our work and other activity recognition, this work is mainly distinguishing the similar activities which designs a new dataset and adds duration and time block features. Comparing the existing research, we adopt MLN with time, location and time sequence features. These features are not easy to express by other algorithms. Therefore, we make the comparison experiments with the MLN (without adding duration and time block features) which has a good performance in interleaved

and concurrent complex situations. The comparison results are given details in following parts.

Alchemy 2.0 is an inference engine of MLN. We use Alchemy 2.0 to learn the weight of rule and inference the likelihood probability. The FoL rules have been stored as ".mln" file. The train data with labels has been stored as ".db" file. The test data has also been stored as " .db" file.

5.1 Similar activity recognition

The duration is a unique habit for residents, because of the essence difference between similar activities, like Boiled Dishes and Fried Dishes, the duration is obvious different. In addition, the time block is one of a typical and different habits in similar activities, different time block has different activity preference, like Drink Coffee and Drink Tea, residents usually choose the different time block in one day. We can test our ideas by the following experiments. For the FriedDishes and BoiledDishes two similar activity groups, the recognition accuracy ratings have been improved from an average 90.3% (the result is shown in Fig. 3) to 92.5%
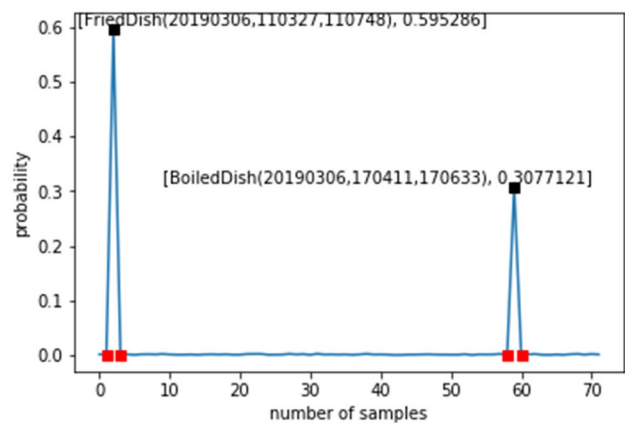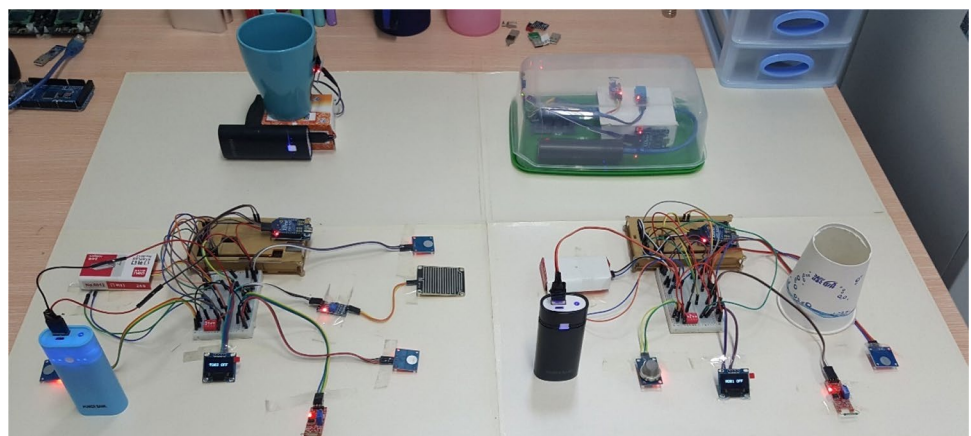

Fig. 3 *Frieddishes* and *Boileddishes* recognition result time series and location only

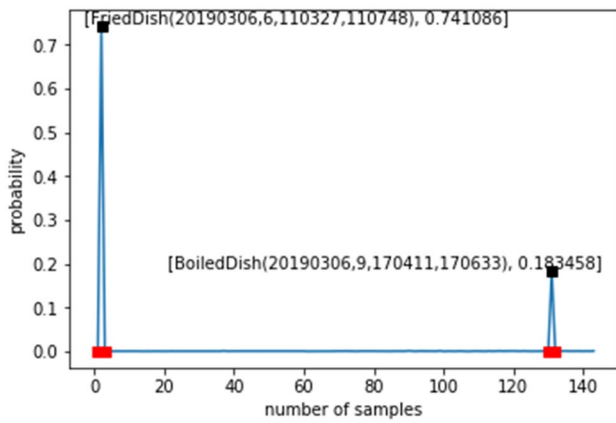**Fig. 2** Simulation deployment diagram for Kitchen Room

Fig. 4 *Frieddishes* and *Boileddishes* recognition result with additional duration and time block

Table 6 Two similar activities probability

Activities group	Probab. (before)	Probab. (modify)
Drinktea and *Drinkcoffee*	0.846	0.910
Washface and *Washcloth*	0.115	0.866
Havemeal and *Dodishes*	0.028	0.992
Drinkmilk and *Drinkjuice*	0.215	0.983
Frieddishes and *Boileddishes*	0.903	0.925
Sweep and *Wipe*	0.842	0.875

(the recognition result is shown in Fig. 4). The horizontal axis represents all the possible activity results when some sensors are triggered by two similar activities (FriedDishes and BoiledDishes). The vertical axis represents the probability of these possible activity results. The right results mean the begin time and end time are right. Adding duration and time block features, the total probability of two similar activities has been improved. From those two figures, we can

Table 7 Father nodes model operation time comparison

Activities group	Oper. time (before)	Oper. time (after)
Drink	76 min 41 s	32 min 27 s
Wash	39 min 33 s	10 min 22 s
Meal	535 min	403 min 16 s
Cook	20 min 24 s	4 min 15 s
Clean	78 min 47 s	15 min 48 s

easily find that the accuracy increases, and the error in the result disappeared. The results for the six similar activities groups are shown in Table 6. The second column "Probab. (before)" is the probability without the duration and time block features, and the third column "Probab.(modify)" adds the duration and time block features whose performance has been improved.

5.2 Hierarchical structure model

For the fivr typical activities categories, Activity (*Drink*), Activity (*Wash*), Activity (*Meal*), Activity (*Cook*), Activity (*Clean*), which are the sets of objects. Sub activities belong to the father nodes, and the related rules of these father nodes have been inherited. Inference processing builds the instances graph which is times bigger than none father nodes network just for the related rules. MLN is based on the rule to construct the related max fully connected subgraph. When we extract the father node from those sub activities, one subgraph has been segmented to two subgraphs and another new concept is added to express the father node.

In order to avoid the computation cost, we preprocess the inheritance which retains the structuration advantages and reduces the inference complexity. The preprocessing pseudo-code to construct the non-father nodes rules has been described as following:

Algorithm 2 Delete father Nodes

1: **function** DELETE(F, L, C) ▶ Where F - rule, L - Logic of MLN, C - Constant of MLN
2:     $G \leftarrow F$
3:     **while do** $F \neq \emptyset$
4:         **for** $q \in F$ **do**
5:             **if** q$\in FathernodesSet$ **then**
6:                 $F \leftarrow F \cup (SonOf(q)\backslash q)$
7:             **end if**
8:         **end for**
9:     **end while**
10:    Return F
11: **end function**

We have compared the operating time with traditional Markov Logic Network and adding the "Delete Father Nodes" algorithm, we can find the operating time has been decreased. The experiment results have been shown in Table 7.

6 Conclusion

This paper focused on improving the performance of similar activity recognition. We have presented two new characteristics which can restrict activity inference rules to improve reasoning efficiency. We introduce duration and time block of an activity to sensor data with time series and location to expand inference rules. We can easily recognize the similar activity which happened at the same day. Research findings have shown, based on similar activities to generalize the hierarchical activity models enhances expandability and readability. In order to decrease the computation cost of adding father node to MLN, we have proposed a preprocessing method to reduce the complexity.

The solution of this paper can be generalized to other field, especially for the timely and accuracy personalized service areas. For future work, other high dimension characteristics should be considered for activity modelling which can get more accuracy representation and inference.

Compliance with ethical standards

References

Brostow GJ, Shotton J, Fauqueur J, Cipolla R (2008) Segmentation and recognition using structure from motion point clouds. In: Forsyth D, Torr P, Zisserman A (eds) Computer vision—ECCV 2008. Springer, Berlin, pp 44–57

Chahuara P, Fleury A, Portet F, Vacher M (2012) Using markov logic network for on-line activity recognition from non-visual home automation sensors. In: Paternò F, de Ruyter B, Markopoulos P, Santoro C, van Loenen E, Luyten K (eds) Ambient intelligence. Springer, Berlin, pp 177–192

Chamroukhi F, Mohammed S, Trabelsi D, Oukhellou L, Amirat Y (2013) Joint segmentation of multivariate time series with hidden process regression for human activity recognition. Neurocomputing 120:633–644

Chen L, Nugent C (2009) Ontology based activity recognition in intelligent pervasive environments. Int J Web Inf Syst 5(4):410–430

Cook DJ, Crandall AS, Thomas BL, Krishnan NC (2013) Casas: a smart home in a box. Computer 46(7):62–69

Domingos P, Lowd D (2009) Markov logic: an interface layer for artificial intelligence. Synth Lect Artif Intell Mach Learn 3(1):1–155

Domingos P, Kok S, Lowd D, Poon H, Richardson M, Singla P (2008) Markov logic. Probabilistic inductive logic programming. Springer, Berlin, pp 92–117

Duong TV, Phung DQ, Bui HH, Venkatesh S (2006) Human behavior recognition with generic exponential family duration modeling in the hidden semi-Markov model. In: 18th International Conference on Pattern Recognition (ICPR'06). IEEE, Hong Kong, China, pp 202–207

Duong T, Phung D, Bui H, Venkatesh S (2009) Efficient duration and hierarchical modeling for human activity recognition. Artif Intell 173(7):830–856

Gao L, Bourke A, Nelson J (2014) Evaluation of accelerometer based multi-sensor versus single-sensor activity recognition systems. Med Eng Phys 36(6):779–785

Gayathri KS, Elias S, Ravindran B (2015) Hierarchical activity recognition for dementia care using markov logic network. Person Ubiquitous Comput 19(2):271–285

Gayathri K, Easwarakumar K, Elias S (2017) Probabilistic ontology based activity recognition in smart homes using markov logic network. Knowl Based Syst 121:173–184

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

Hooda D, Rani R (2020) Ontology driven human activity recognition in heterogeneous sensor measurements. J Ambient Intell Humanized Comput. https://doi.org/10.1007/s12652-020-01835-0

Junhuai L, Ling T, Huaijun W, Yang A, Kan W, Lei Y (2019) Segmentation and recognition of basic and transitional activities for continuous physical human activity. IEEE Access 7:42565–42576

Liu L, Peng Y, Liu M, Huang Z (2015) Sensor-based human activity recognition system with a multilayered model using time series shapelets. Knowl Based Syst 90:138–152

Liu L, Cheng L, Liu Y, Jia Y, Rosenblum DS (2016) Recognizing complex activities by a probabilistic interval-based model. In: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI Press, AAAI'16, Phoenix, Arizona, USA, pp 1266–1272

Machot FA, Mayr HC, Ranasinghe S (2016) A windowing approach for activity recognition in sensor data streams. In: 2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN). IEEE, Vienna, Austria, pp 951–953

Meditskos G, Dasiopoulou S, Efstathiou V, Kompatsiaris I (2013) Spact: a hybrid framework for complex activity recognition combining owl and sparql rules. In: 2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops). IEEE, San Diego, CA, USA, pp 25–30

Moutacalli T, Bouchard B, Bouzouane A (2015) The behavioral profiling based on times series forecasting for smart homes assistance. J Ambient Intell Hum Comput 6(5):647–659

Nazerfard E (2018) Temporal features and relations discovery of activities from sensor data. J Ambient Intell Humanized Comput. https://doi.org/10.1007/s12652-018-0855-7

Okeyo G, Chen L, Wang H (2014a) Combining ontological and temporal formalisms for composite activity modelling and recognition in smart homes. Future Gen Comput Syst 39:29–43

Okeyo G, Chen L, Wang H, Sterritt R (2014b) Dynamic sensor data segmentation for real-time knowledge-driven activity recognition. Perv Mob Comput 10:155–172

Ordóñez FJ, De Toledo P, Sanchis A (2013) Activity recognition using hybrid generative/discriminative models on home environments using binary sensors. Sensors 13(5):5460–5477

Pansiot J, Stoyanov D, McIlwraith D, Lo BP, Yang GZ (2007) Ambient and wearable sensor fusion for activity recognition in healthcare monitoring systems. In: Leonhardt S, Falck T, Mähönen P (eds) 4th international workshop on wearable and implantable body sensor networks (BSN 2007). Springer, Berlin, pp 208–212

Riboni D, Bettini C (2011) Cosar: hybrid reasoning for context-aware activity recognition. Person Ubiquitous Comput 15(3):271–289

Ryoo MS, Aggarwal JK (2009) Semantic representation and recognition of continued and recursive human activities. Int J Comput Vis 82(1):1–24

Saguna S, Zaslavsky A, Chakraborty D (2013) Complex activity recognition using context-driven activity theory and activity signatures. ACM Trans Comput Hum Interact 20(6):1–34

Singla P, Domingos P (2005) Discriminative training of markov logic networks. In: Proceedings of the 20th National Conference on Artificial Intelligence, vol 2, AAAI Press, AAAI'05, Pittsburgh, Pennsylvania, USA, pp 868–873

Suryadevara NK, Mukhopadhyay SC (2014) Determining wellness through an ambient assisted living environment. IEEE Intell Syst 29(3):30–37

Tahir SF, Fahad L, Kifayat K (2019) Key feature identification for recognition of activities performed by a smart-home resident. J Ambient Intell Humanized Comput. https://doi.org/10.1007/s12652-019-01236-y

Tran SD, Davis LS (2008) Event modeling and recognition using markov logic networks. In: Forsyth D, Torr P, Zisserman A (eds) Computer vision—ECCV 2008. Springer, Berlin, pp 610–623

Yan S, Liao Y, Feng X, Liu Y (2016) Real time activity recognition on streaming sensor data for smart environments. In: 2016 International Conference on Progress in Informatics and Computing (PIC). IEEE, Shanghai, China, pp 51–55

Ye J, Stevenson G, Dobson S (2015) Kcar: a knowledge-driven approach for concurrent activity recognition. Perv Mob Comput 19:47–70

Zhang S, McClean S, Scotney B, Chaurasia P, Nugent C (2010) Using duration to learn activities of daily living in a smart home environment. In: 2010 4th International Conference on Pervasive Computing Technologies for Healthcare. IEEE, Munich, Germany, pp 1–8

Zhang Y, Zhang Y, Swears E, Larios N, Wang Z, Ji Q (2013) Modeling temporal interactions with interval temporal bayesian networks for complex activity recognition. IEEE Trans Pattern Anal Mach Intell 35(10):2468–2483

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