International Journal of Hybrid Information Technology
Vol.9, No.4 (2016), pp. 41-50
http://dx.doi.org/10.14257/ijhit.2016.9.4.04

A New Method for Context Prediction based on Markov Logic Network in Specific Situations

Lu Fei, Li Jian

School of Control Science and Engineering, Shandong University, Jinan, 250061, China
lawyerlf@sdu.edu.cn

Abstract

Context information in smart space are complex and uncertain, it increases the difficulty of information prediction. In order to increase the accuracy of information prediction, a new method is proposed in this paper. Firstly, first-order predicate logic is used to describe the context information and Markov logic network is used to achieve the unification of the first-order logic and probabilistic graphical models. Secondly Markov logic network model is established based on some particular scenes which were predicted by neural network, the purpose of this model is to realize the prediction of device information which is valuable for the task planning by relevant complex context information. At last, simulation test is done using the real data collected from our smart space laboratory, the experiment results show that method is effective.

Keywords*: smart space, context information, markov logic network, first-order predicate logic, information processing.*

1. Introduction

Smart space is a smart area equipped with a variety of smart sensors, devices and many service equipments. The concept of smart space was firstly proposed by Hashimoto Laboratory at the University of Tokyo in 1995[1]. The main function of smart space is to obtain context information from the distributed sensors and to provide support information for corresponding tasks. But in most cases, the context information are uncertain and incomplete. These characteristics will affect the accuracy of the decision-making. In order to solve the influence to information reasoning caused by the above characteristics, scholars have proposed a variety of solutions, such as: the context toolkit was proposed by Dey and Abowd in Georgia[2]. This solution draws on the thinking of GUI and abstracts Widget, Interpreter and Aggregator three parts. During the execution, they used the way of class library and achieved the expectation. Another scheme was proposed by Jonsson of KTH in Sweden, and this scheme is similar to the first one, but it exists in the form of server independent of applications. The novel place of this scheme lies in the obtainment of context which is called Context Repository[3]; Additional, two solutions: Context Fabric[4] and Context Logic[5] were proposed. The first one was proposed by Hong in Berkeley and the second was proposed in Tsinghua university. Both of them introduced the context query language and reasoning model to deal with context information, they can deal with a bigger amount of data and minority uncertain information to some extent. These proposed schemes can solve a certain degree of uncertainty and reasoning requirements, and even they have taken into account the context query information. But the downside is that they did not consider the complexity of the context information and the integrity of the reasoning model.

In order to deal with the context information in smart space and improve the accuracy of information reasoning, a novel proposal is proposed in this paper. Firstly, first-order predicate logic is used to describe the context information and Markov logic network is

ISSN: 1738-9968 IJHIT

used to achieve the unification of the first-order logic and probabilistic graphical models. Secondly Markov logic network model is established based on some particular scenes which were predicted by Neural network, the purpose of this model is to realize the prediction of device information which is valuable to the task planning by relevant complex context information. At last, simulation test is done using the real data collected from our smart space laboratory, the results of experiment show that method is effective.

2. First-order Logic Description of Context Information

A first-order knowledge base is a collection of logic sentences and rules. The rules are composed of four types of symbol: constant, variable, functions and the predicates. Constant symbols represent objects in some fields (such as: device: curtain, light, air-condition and so on). The variable symbols can change its value. The function symbols represent the mapping relationship between the groups of objects (such as: meeting (Light) (The meaning is that this light is on in the meeting room)). Predicate symbols represent the relationship between the objects (for example: House (x, y) (x and y are equipments and they are in the same room)) or the object's properties (such as: Open (Light) (light is open)).Moreover predicate symbols also should represent that which symbols stand for the responding objects, functions and relations in a certain domain. Variables and constants can have a type, usually constant can only represent the objects of the same type. Variables can have a changeable value or object in a range of number, for example, the variable x can represent devices (such as curtain, electric light, etc.), the constant C can stand for a space (for example: Space1) [6]and so on.

In this paper, the form such as House (Curtain, the meeting (Light)) is called first-order logic, and we can follow this method. In the below, the concept of weight is introduced to combine the first-order predicate logic with probability. In order to realize knowledge base maintenance conveniently, the first-order predicate logic is expressed in the disjunctive mode. The context information is shown in table 1.

Table 1. Description of Context in First-order Logic

Description	Meeting cause device Open	If two devices are in the same House, states of them are same	
first-order logic	∀ x meeting(x)⇒Open(x)	∀ x∀ y Hos (x,y)⇒(Open (x)⇔Open (y))	
canonical form	¬ meeting (x)∨Open (x)	¬ Hos (x,y)∨Open (x)∨¬ Open (y)	¬ Hos (x,y)∨¬ Open (x)∨Open (y)
weight	1.3	1.0	1.0

3. Markov Logic Network

Markov logic network(MLN: Markov logic network) is a kind of statistical correlation method, and the context information in the smart space mostly has the property of correlation. The prediction result to the context information in smart space will be better with the using of Markov logic network. Markov logic network is composed by Markov network (MRF: Markov random field) and the first-order logic.

3.1. Markov Network

Markov network which also called Markov random field (MRF) is a joint distribution model including random variable $x = x_1, x_2, ..., x_n$. It is shown in figure 1.

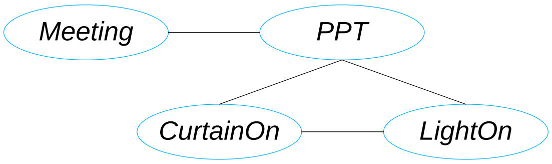


Figure 1. Undirected Graphical Model

In figure 1, the states of three devices and meeting are included in the Markov network. It is composed of an undirected graph *G* and a set of potential functions Φ_k. Each variable is seen as a node on the graph *G* and potential functions Φ_k are defined over cliques. Potential function is a non-negative real function, it represents the real state of the clique. The joint distribution represented by a Markov network is given as followed.

$$P(x) = \frac{1}{Z}\prod_c \Phi_c(x_c) \qquad (1)$$

$$Z = \sum_x \prod_c \Phi_c(x_c) \qquad (2)$$

We can see that the *Z* is the partition function. In order to use the weight directly, linear the equation(1),the result is given as followed.

$$P(x) = \frac{1}{Z}\exp\left(\sum_i w_i f_i(x)\right) \qquad (3)$$

Where, w_i is the weight of variable *i* and $f_i(x)$ is the feature function of variable *i*. This feature function can be any real-valued function, In order to calculate and said conveniently, this paper regulates that this function is a binary function[7].

3.2. Definition of Markov Logic Network

Based on the above knowledge, we can build a first-order logic knowledge base, the rules in the knowledge base are a series of hard constraints. So the purpose of building Markov logic network is to soften these constraints. When a world violates one formula in the knowledge base, it is less probable, but not impossible. The less formulas a world violates, the more probable it is. Each formula has an associated weight that reflects how strong a constraint it is: the higher the weight, the greater the log probability difference is between a world that satisfies the formula and not. The detailed definition is listed as below.

Definition: A Markov logic network M is a set of pairs (F_i, w_i), where F_i is a formula in first-order logic and w_i is a real number. Together with a finite set of constants C, it defines a Markov network $L_{M,C}$.

(1) Each ground atom in M corresponds to a binary node in $L_{M,C}$.The value of the node is 1 if the ground atom is true, otherwise it is 0.

(2) $L_{M,C}$ contains one feature for each possible grounding of each formula F_i in M. The value of this feature is 1 if the ground formula is true, otherwise it is 0. The weight of the feature is the w_i associated with F_i in M. From the definition and equation (1) and (2), Markov network $L_{M,C}$ is given by

$$P(X=x)=\frac{1}{Z}\exp\{\sum_i w_i n_i(x)\}=\frac{1}{Z}\prod_i \Phi_i(x_i)^{n_i(x)} \qquad (4)$$

Where $n_i(x)$ is the number of true groundings of F_i, x_i is the state (truth values) of the atoms appearing in F_i, and $\Phi(x_i)=e^{w_i}$.

In equation4, we defined Markov logic network as log-linear models in the first equality, the second equality shows the product form of equivalent potential functions.

4. The Establishment and Use of Knowledge Base in Smart Space

4.1. Markov Logic Network Based on the Specific Situation

To reduce the size of the Markov network and improve the prediction accuracy, an establishing method of MLN based on specific situation is proposed in this paper. Due to the environment's complexity and variability in smart space, a lot of contexts in certain situation are unchanging. At same time, user is the serving object in smart space, so the state and number of users also determine the prediction accuracy of the context information. Table 2 describes the classification of particular situations.

Table 2. Classification of Specific Situations

Situations	Situation 1	Situation 2
Users	existence	inexistence
Time	Working	Relaxing
Space	Same	Different

In table2, there are 8 kinds situations from this classification. We can establish different Markov logic network in different situations with the same context formulas and we will also get different results. Next we will build a Markov network with the situation which exists people, working time and the same space. From figure2: (House (C, L) stands for that C and L are in the same House, Open (C) stands for the open state of equipment C and Meeting means this space is at a meeting state.)

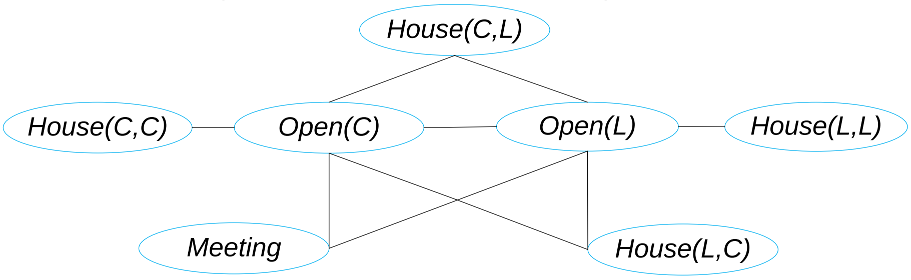


Figure 2. Model of Markov Network

It is formed by the two rules in table 1 and the constants: curtain and light. Each node in the graph is the basic atoms (such as, House (C, L)). The graph contains an arc between each pair of atoms if they appear together in a basic rule.. It has three cliques which has three atoms. This Markov network can infer the possibility of the meeting according to the status of curtain and light when someone is in this space at working time or infer the state of curtain and light based on the fact that this space is meeting, and so on.

4.2. Parameter Learning and Prediction of Markov Logic Networks

After determining the structure of MLN, the main task is to conduct parameter learning and modify the weights in the network based on real data. So the main purpose of learning is to adjust the weights of parameters. This paper follows the literature [8] on parameter learning with pseudo maximum likelihood estimation. In literature [9], the function of pseudo maximum likelihood estimates can be described as:

$$P_w^*(X=x)=\prod\nolimits_{i=1}^{n} P_w(X_l=x_l \mid MB_x(X_l)) \tag{5}$$

The gradient of the pseudo-log-likelihood is

$$\frac{\partial}{\partial w_i}\log P_w^*(X=x)$$
$$=\sum_{l=1}^{n}[n_i(x)-P_w(X_l=0\mid MB_x(x_l))n_i(x_{[X_l=0]})-P_w(X_l=1\mid MB_x(X_l))n_i(x_{[X_l=1]})] \tag{6}$$

Where, x_l is the number of true groundings of the *l* th formula, and $MB_x(X_l)$ is the state of the Markov blanket of x_l in the data. Computing this expression (or Eq. (5)) does not require inference over the model. We optimize the pseudo-log-likelihood using the limited-memory BFGS algorithm based on literature [8].

To realize Markov logic network inference, we calculate marginal probability and the conditional probability. That is, when the structure and constants of Markov logic network are determined, we take the true probability of a rule as marginal probability, and when calculating, we have given some value of other rules, so we call this as condition probabilities. When given a Markov logic network M, constant set of T, and the value of a given rule F2, the calculated formula of rule F1 is :

$$P(F_1 \mid F_2,M,T)=\frac{P(F_1 \wedge F_2 \mid L_{M,T})}{P(F_2 \mid L_{M,T})}=\frac{\sum_{X\in\theta_{F_1}\cap\theta_{F_2}} P(X=x\mid L_{M,T})}{\sum_{X\in\theta_{F_2}} P(X=x\mid L_{M,T})} \tag{7}$$

Where, θ_F is the set of worlds where *F* holds true, and $P(F_2 \mid L_{M,T})$ is the probabilities of all the worlds which can let F_2 be true. In this paper, the calculation to unknown rules is done through the MCMC algorithm and the first-order predicate logic optimization strategy in literature [10][11].

5. The Application of Markov Logic Network in Specific Situation

5.1. Determination of Certain Circumstances in Smart Space

In order to reduce the size of Markov logic network, this article will divided smart space environment into specific situations depending on the context of the information, and use neural network algorithm to accurately estimate the situation based on collected information. Seen from the above, the application scenarios of Markov logic network are divided into eight.

According to the different types of context, a three-forward neural network structure which includes 6 inputs and 8 outputs is designed. The structure is shown in figure 3.

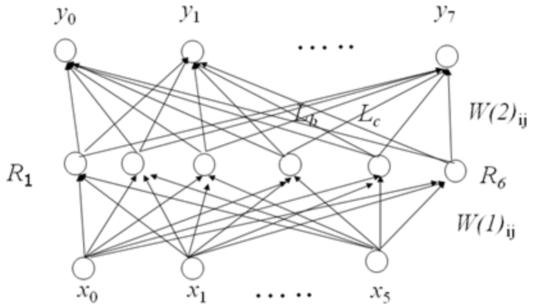


Figure 3. Structure of Designed Neural Network

In reality, we collect the context information in each of the eight situations based on the neural network. We divided the context into six types: temperature, location, intensity, noise, action, relationship as the input of neural network, and use the real results as output to train the neural network.

After training, the neural network is applied to each situation. This method can update the weights and determine the specific situation based on context information online.

5.2. Specific Applications of Markov Logic Network in Smart Space

Since we use specific situation to distinguish MLNs, eight corresponding files will be got when a Markov logic network model is established. And each file is corresponding to a specific filename. We use the Markov logic network application platform - Alchemy system to complete the prediction task under specific situations. This platform is developed at the University of Washington. The main steps are as follows:

Step1: Due to the difference of service tasks in smart space, the tasks are divided into: office tasks, living tasks, urgent tasks and showing tasks.

Step2: In each identified smart environment, the monitored context information is classified into: temperature class, location class, intensity class, noise class, action class, relationships class and so on.

Step3: Create Markov logic network according to the needs of different tasks, i.e. create mln file in Alchemy system. In this step we do not need to consider the specific circumstances, we just need to establish an unified mln file.

Step4: Collect training data according to the eight particular circumstances. Collected context information at eight different contexts and build eight training data files for training.

Step5: Use Alchemy system to train Markov logic network. By training system, we will get eight different output files for specific situations. The detailed way is that we use the command ‘learnwts’ in Alchemy system to learn weights, and get .mln files which contain formula weights.

Step6: According to the relevant contextual information in Step 2, we use a neural network to determine the situation in smart environment, and select the appropriate .mln file for reasoning.

Step7: The transplant of the inference algorithm. Due to the programming language and operating system of Alchemy system, we can’t apply this Alchemy system directly to our own system. So we use Windows platform and C++ to transplant l Alchemy system to Windows platform. The algorithm will be programmed based on marginal probability and conditional probability, and the test data acquisition and related information will be marked on the timestamp during the reasoning process. The corresponding Markov logic network is determined according to the information related to specific situation

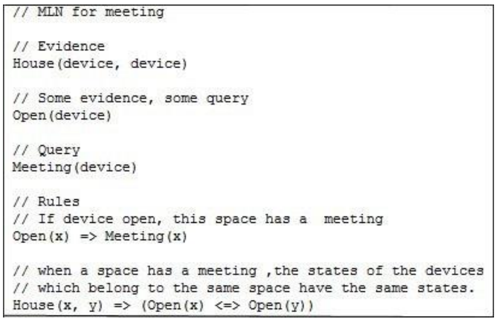
Step8: Compare the output results and output predictive information.

6. The Experimental Results

6.1. Construction of the Original Markov Logic Network and Parameter Training

6.1.1. Construction of Original Network

This experiment is done to test the application of Markov logic network under certain circumstance in the smart space. A smart meeting task is used for the test. According to the status of related equipment in each conference room, the new system predict the real statue of these rooms. Then the Markov model is created and is shown in figure 4. The mln file used by Alchemy system is as follows.

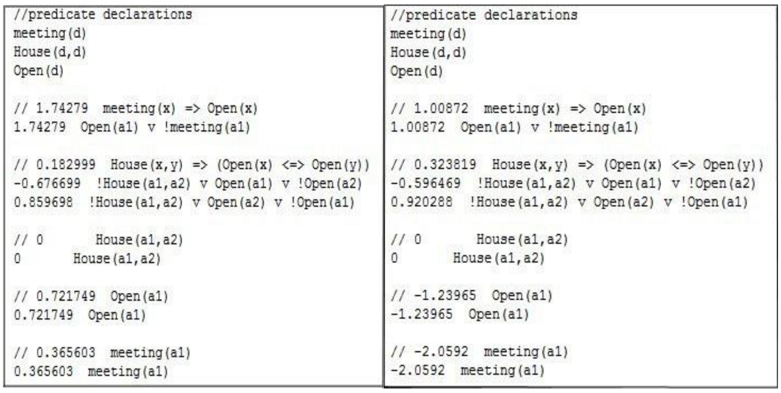


```
// MLN for meeting

// Evidence
House(device, device)

// Some evidence, some query
Open(device)

// Query
Meeting(device)

// Rules
// If device open, this space has a  meeting
Open(x) => Meeting(x)

// when a space has a meeting ,the states of the devices
// which belong to the same space have the same states.
House(x, y) => (Open(x) <=> Open(y))
```

Figure 4. Data of Original Markov Model

6.1.2. Parameter Training

For parameter training, the training data should be got firstly. According to the specific requirements of the situation, in our smart space laboratory, we start smart meeting room mode and collect relevant context data at all situations even there are no meetings. At last, we get 8 training data files and store these files to .Db files. Two of the data formats are shown in figure 5.



```
//predicate declarations
meeting(d)
House(d,d)
Open(d)

// 1.74279  meeting(x) => Open(x)
1.74279  Open(a1) v !meeting(a1)

// 0.182999  House(x,y) => (Open(x) <=> Open(y))
-0.676699  !House(a1,a2) v Open(a1) v !Open(a2)
0.859698  !House(a1,a2) v Open(a2) v !Open(a1)

// 0       House(a1,a2)
0       House(a1,a2)

// 0.721749  Open(a1)
0.721749  Open(a1)

// 0.365603  meeting(a1)
0.365603  meeting(a1)
```

```
//predicate declarations
meeting(d)
House(d,d)
Open(d)

// 1.00872  meeting(x) => Open(x)
1.00872  Open(a1) v !meeting(a1)

// 0.323819  House(x,y) => (Open(x) <=> Open(y))
-0.596469  !House(a1,a2) v Open(a1) v !Open(a2)
0.920288  !House(a1,a2) v Open(a2) v !Open(a1)

// 0       House(a1,a2)
0       House(a1,a2)

// -1.23965  Open(a1)
-1.23965  Open(a1)

// -2.0592  meeting(a1)
-2.0592  meeting(a1)
```

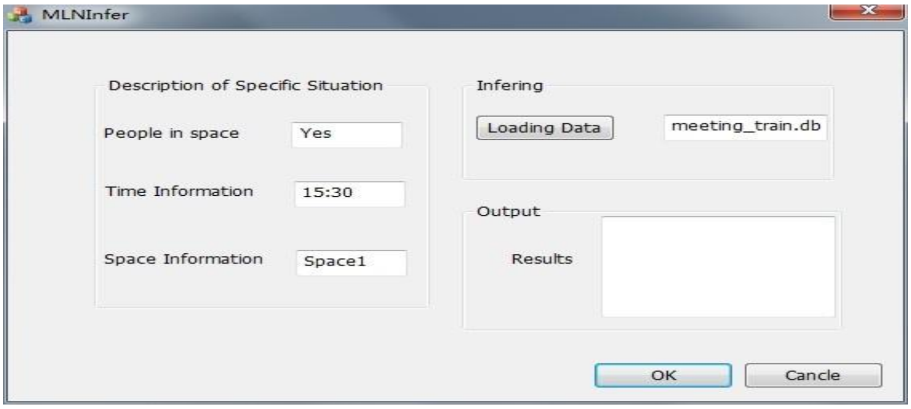
Figure 5. Output Model Data of Training

From the above results, the training data is different under different situations. After getting the 8 kinds of training data from one original Markov logic network model, we add timestamp and attribute information to them. The format is <time|person|space>.

6.2. Platform of Prediction Algorithm and Reasoning

6.2.1. Transplant of Algorithm

In our smart space environment, information collection and information storage are based on the windows platform. In order to keep the system for real time working and complete the proposed requirements of reasoning under certain circumstances, we use VS2012 to achieve the prediction algorithm transplanting from Alchemy system. The transplant procedure is shown in figure 6.



MLNInfer

Description of Specific Situation

People in space: Yes

Time Information: 15:30

Space Information: Space1

Infering

meeting_train.db

Output

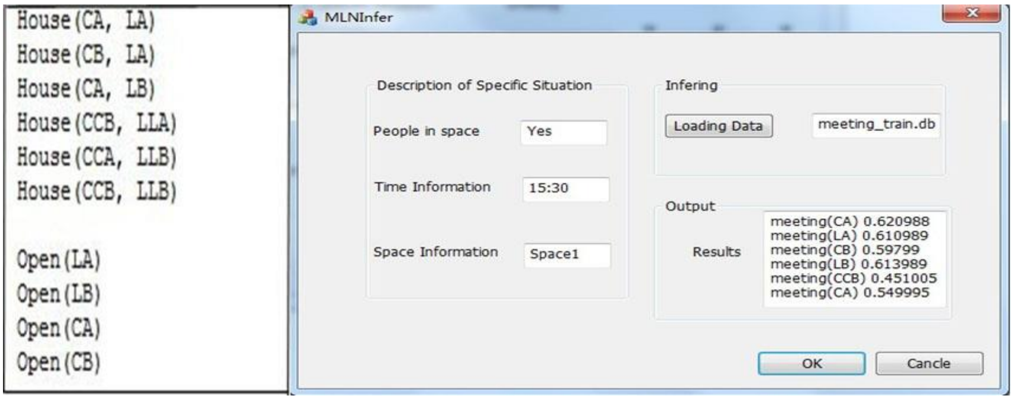
Results:

Figure 6. Testing Transplant Procedure

Where, the description of specific situation is inferred by the neural network algorithm according to the context information. Validated by the experimental data and real environment, the neural network prediction accuracy for specific situations is up to 95%. Neural network algorithm greatly improves the ability of the new system to predict automatically.

6.2.2. Reasoning Results

According to the algorithm after transplantation, the different testing data is loaded into the algorithm. We get the outputting result in the form of calculating probability calculation and marginal probability. At last we find the valuable information for mission planning. The output test data and reasoning data is in figure 7.



```
House(CA, LA)
House(CB, LA)
House(CA, LB)
House(CCB, LLA)
House(CCA, LLB)
House(CCB, LLB)

Open(LA)
Open(LB)
Open(CA)
Open(CB)
```

MLNInfer

Description of Specific Situation

People in space: Yes

Time Information: 15:30

Space Information: Space1

Infering

meeting_train.db

Output

Results

```
meeting(CA) 0.620988
meeting(LA) 0.610989
meeting(CB) 0.59799
meeting(LB) 0.613989
meeting(CCB) 0.451005
meeting(CA) 0.549995
```

Figure 7. Reasoning Results

With the output results, the inference algorithm can accurately predict state of a meeting room according to the corresponding switch state of curtains and lights. The final result will be output by comparing the output probability and designed threshold. That is, Space1 holds a meeting based on the judgment.

6.3. Comparison of Markov Logic Network and Bayesian Network

Through the above content, after the establishment of a Markov logic network based on specific situations, we can predict the context information in smart space. The advantage of Markov logic network is that the scale of MLN is small and the speed of the prediction can be improved. So in the point of network scale and speed of prediction, this paper uses Bayesian network to compare with Markov network. The comparative results are given as followed.

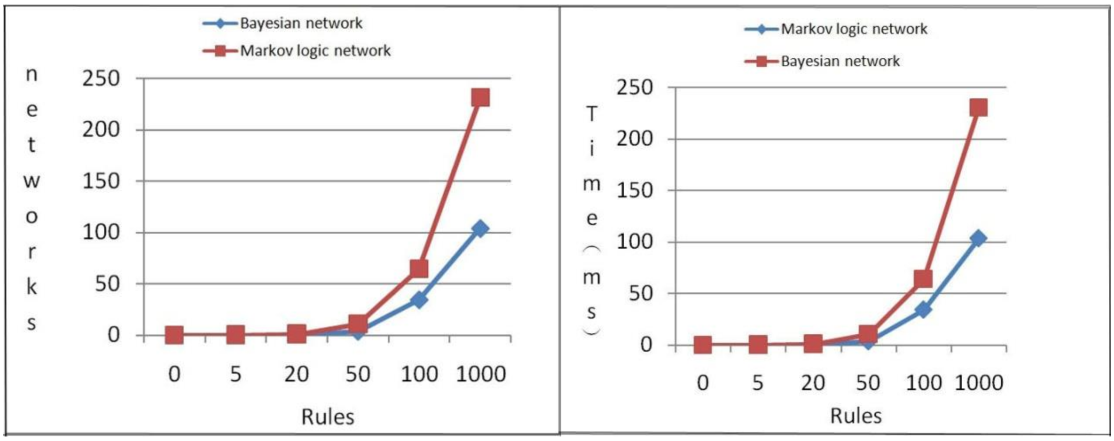


Figure 8. Markov Logic Network vs. Bayesian Network

We can see from the results when the rules are same, the number of Bayesian network is obviously less than the number of Markov logic network. This is due to the shrinking of Markov logic network. However, under the same rules, as the increase in the number of rules, the predicted time of Markov logic network is less than that of Bayesian network with the increasing of rules. It is the requirement of the real-time forecast in smart space. So the effect of the Markov logic network applied to the smart space is better.

7. Summary

Based on the complexity and uncertainty of the context in smart space, this paper proposed a way of using first-order predicate logic to handle the context information in smart space and established Markov logic network under specific situations to predict relevant contextual information. The processing to complex context information in smart space is achieved. This method provides the driving data for service planning in smart space, and it is also a key solution to subsequent service task scheduling research. Next work will be focused on a full transplant of alchemy system and the establishment of Markov logic network with all the finishing context information.

Acknowledgement

This work was supported by National Natural Science Foundation of China (NO.61305122) and Postdoctoral Science Foundation of China (2014M561928).

References

[1] J. H. Lee, N. Ando, H. Hashimoto, “Design policy of intelligent space”, Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, **(1999)**, Tokyo, Japan.

[2] A K. Dey, “Providing architectural support for building context-aware applications”, Georgia Institute of Technology, **(2001)**.

[3] J. I. Hong, “An infrastructure approach to context-aware computing”, Human-computer interaction, vol.16, no.2, **(2002)**, pp.287-303.

[4] J. I. Hong, J. A. Landay, “An infrastructure approach to context-aware computing”, Human-Computer Interaction, vol.16, no.2, **(2001)**, pp.287-303.

[5] Xie W K, “Key support technologies of smart space”, Tsinghua University, **(2003)**.

[6] M. Richardson, P. Domingos, “Markov logic networks”, Machine learning, vol. 62, no.1, **(2006)**, pp.107-136.

[7] C. F. Xu, “ChunL Hao, BaoJ Sun”, Research of markov logic network, Journal of Software, vol.22, no.8, **(2011)**, pp.1699-1713.

[8] P. Domingos, M. Richardson, “Markov Logic: A unifying framework for statistical relational learning”, Proceedings of the workshop on Statistical Relational Learning and its connections to other fields, **(2004)**; Banff, Canada.

[9] D. C. Liu, J. Nocedal, “On the limited memory BFGS method for large scale optimization”, Mathematical programming, vol.45, no.1, **(1989)**, pp. 503-528.

[10] W.R. Gilks, S. Richardson, D.J. Spiegelhalter, “Markov chain Monte Carlo in practice”, CRC press, **(2005)**.

[11] S. Kok, P. Domingos, “Learning the structure of Markov logic networks”, Proceedings of the 22nd international conference on Machine learning, **(2005)**; Germany.

Authors

Lu Fei. She was born in 1973. She received the Ph.D. degree from Shandong University. Currently, she is an associate professor in school of Control Science and Engineering, Shandong University, China. Her research interests include intelligent space technology, service robot and intelligent control.