

# Mining Sensor Data in Smart Environment for Temporal Activity Prediction

Vikramaditya Jakkula  
Washington State University  
EME 206, Spokane Way  
Pullman, WA 99164  
vikramaditya@wsu.edu

Diane J. Cook  
Washington State University  
EME 206, Spokane Way  
Pullman, WA 99164  
cook@eecs.wsu.edu

## ABSTRACT

Technological enhancements aid development and advanced research in smart homes and intelligent environments. The temporal nature of data collected in a smart environment provides us with a better understanding of patterns over time. Prediction using temporal relations is a complex and challenging task. To solve this problem, we suggest a solution using probability based model on temporal relations. Temporal pattern discovery based on modified Allen's temporal relations [8] has helped discover interesting patterns and relations on smart home datasets [17]. This paper describes a method of discovering temporal relations in smart home datasets and applying them to perform activity prediction on the frequently-occurring events. We also include experimental results, performed on real and synthetic datasets.

## Keywords

Sensor Data, Knowledge Discovery, Temporal Relations, Smart Environments, Prediction.

## 1. INTRODUCTION

The need for an enhanced prediction model is essential for any intelligent smart home to function in a dynamic world. For an agent to perform activity prediction, it should be capable of applying the limited experience of environmental event history to a rapidly changing environment, where event occurrences are related by temporal relations. Temporal rule mining has been attracting some considerable attention over the decade [5] [16]. In this paper we consider the problem of activity prediction based on discovery and application of temporal relations. Prediction with higher accuracy can be of great importance to a smart environment. For instance consider a scenario, where we have a temporal analysis based predictor in a smart environment, which adapts to the inhabitant, who has an habit of cooking turkey when he watches television for greater than five hours, then we see that this temporal based predictor compared to a sequential prediction would predict the use of oven later that day and can also be helpful for reminder assistance and also maintenance issues which include whether there is turkey at home before he just walks into the kitchen to notice the shortage of turkey.

Allen suggested that it was more common to describe scenarios by time intervals rather than by time points, and listed thirteen relations comprising a temporal logic: before, after, meets, meet-by, overlaps, overlapped-by, starts, started-by, finishes, finished-by, during, contains, equals (displayed in Table 1.) [8]. These temporal relations play a major role in identifying temporal activities which occur in a smart home. Consider, for instance, a case where the inhabitant turns the Television (TV) after sitting on the couch. We notice that these two activities, turning on the TV and sitting on the couch, are frequently related in time according to the "after" temporal relation. Therefore, when inhabitant is sitting on a couch, the smart environment can predict that in the near future the television would likely be turned on. From Allen's original thirteen temporal relations [8] we represent and identify the nine relations shown in Table 1. These temporal relationships relate a particular event with the next observed event, and thus are useful for event prediction. Four of Allen's defined temporal relations (before, contains, overlaps, and meets) are not included in this analysis because they do not aid prediction. To analyze smart home data, we first identify temporal relations that occur among events in the data and then apply association rule mining to focus on the event sequences and temporal relations that occur frequently in the smart home. Based on the identified relationships, we calculate the probability of the event most likely to occur.

A question may arise as to why Allen's temporal relations should be used for generating temporal intervals. The temporal relations defined by Allen form the basic representation of temporal intervals, which when used with constraints become a powerful method of expressing expected temporal orderings between events in a smart environment. In addition, they have an easy naming convention, making it easier to recognize, interpret and use the temporal relations that are identified. In earlier work, we performed this prediction based solely on the sequence of observed activities [9]. ActiveLeZi based approach for any given sequence of events that can be modeled as a stochastic process; the algorithm employs Markov models to optimally predict the next symbol in any stochastic sequence. In this work, we supplement evidence for a particular action using the temporal relation information. We illustrate the benefit of temporal relationships for prediction of smart home events. Based on results generated from synthetic and real smart home data, we conclude that temporal logic provides substantial benefits for smart home tasks. Identification of temporal relations provides key insights to smart home activities and aids with prediction and anomaly detection in a smart home or other smart environment. We extend these methods to incorporate valuable information

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ACM KDD '07, August 12–15, 2007, San Jose, California, USA.  
Copyright 2007 ACM X-XXXXX-XX-X/XX/XX ...\$5.00

about the interval of time each event spans. While other methods treat each event as a separate entity (including, for example, turning on a lamp and later turning off the same lamp), our interval-based analysis considers these two events as members of one interval. Each interval is expressed in terms of start time and end time values. As a result, temporal relationships between such intervals can be identified and used to perform critical prediction decisions.

We focus only on our objective to develop a model for analyzing event prediction model in the interval-based events using temporal relations for the most frequently-occurring events, because discovering all possible patterns can be computationally prohibitive in a real world where we have a continuous stream of data which makes the environment dynamic. Furthermore, the number of results obtained can be overwhelming, taxing the ability to effectively use the discovered results for practical purposes. In this paper, we introduce a temporal representation to express relationships between interval-based events. We build on this representation to identify frequently-occurring relationships between temporal events, and use the results as the basis for performing prediction. We explain the temporal relations with illustrations and also include a brief description of the temporal intervals formation process. We describe the steps involved in the experimentation for prediction and present the results obtained on real and synthetic datasets.

## 2. RELATED WORK

All With the converging of supporting technology in artificial intelligence and pervasive computing, smart environment research is quickly maturing. As an example, the MavHome Project treats an environment as an intelligent agent, which perceives the environment using sensors and acts on the environment using sensor network. At the core of its approach, MavHome observes resident activities as noted by the sensors. These activities are mined to identify patterns and compression-based predictors are employed to identify likely future activities [14].

The Gator Tech Smart home is built from the ground up as an assistive environment to support independent living for older people and individuals with disabilities [1]. Currently, the project uses a self-sensing service to enable remote monitoring and intervention caregivers of elderly persons living in the house. The University of Essex's intelligent dormitory (iDorm) is a real ambient intelligent test-bed comprised of a large number of embedded sensors, actuators, processors and networks in the form of a two bed roomed apartment. Fuzzy rules are learned from the observed resident activities [3] and are used to control select devices in the dorm room.

Although smart environments are being designed and created, little related research has focused on reasoning about the timing of events in these environments. We have selected Allen's temporal intervals to represent temporal relationships between events in smart environments. Mörchen argues that Allen's temporal patterns are not robust and small differences in boundaries lead to different patterns for similar situations [11]. Mörchen presented a Time Series Knowledge Representation, which expresses the temporal concepts of coincidence and partial order. Although this method appears feasible and computationally sound, it does not suit our smart home application due to the granularity of the time intervals in smart homes datasets. His

approach does not involve ways to eliminate noise and the datasets are so huge that computational efficiency would not be the only factor to be considered. Björn, et al. [2] also reasons that space and time play essential roles in everyday lives. They discuss several AI techniques for dealing with temporal and spatial knowledge in smart homes, mainly focusing on qualitative approaches to spatiotemporal reasoning.

Ryabov and Puuronen in their work on probabilistic reasoning about uncertain relations between temporal points [19] represent the uncertain relation between two points by the uncertainty vector with three probabilities of basic relations (" $<$ ", " $=$ ", " $>$ "), they also incorporate inversion, composition, addition, and negation operations into their reasoning mechanism. But this model would not be suitable for a smart home scenario as it would not go into more granularity to analyze instantaneous events. Worboys et.al. work [20] involves spatio-temporal based probability models which are currently identified as future work. Dekhtyar et. al. work [18] on probabilistic temporal databases provides a framework which is an extension of the relational algebra that integrates both probabilities and time. This work includes some description of Allen's temporal relations and some of these are incorporated already in this current work.

## 3. ENVIRONMENT SENSING

We define an intelligent environment as one that is able to acquire and apply knowledge about its residents and their surroundings in order to adapt to the residents and meet the goals of comfort and efficiency. These capabilities rely upon effective prediction, decision making, mobile computing, and databases. With these capabilities, the home can control many aspects of the environment such as climate, water, lighting, maintenance, and entertainment. Intelligent automation of these activities can reduce the amount of interaction required by residents, reduce energy consumption and other potential wastages, and provide a mechanism for ensuring the health and safety of the environment occupants [15]. The major goal of MavHome project is to design an environment that acts as an intelligent agent and can acquire information about the resident and the environment in order to adapt the environment to the residents and meet the goals of comfort and efficiency. In order to achieve these goals the house should be able to predict, reason, and make decisions for controlling the environment [9].

MavHome operations can be characterized by the following scenario. To minimize energy consumption, MavHome keeps the house cool through the night. At 6:45am, MavHome turns up the heat because it has learned that the home needs 15 minutes to warm to Bob's desired waking temperature. The alarm sounds at 7:00am, after which the bedroom light and kitchen coffee maker turn on. Bob steps into the bathroom and turns on the light. MavHome records this manual interaction, displays the morning news on the bathroom video screen, and turns on the shower. When Bob finishes grooming, the bathroom light turns off while the kitchen light and display turn on, and Bob's prescribed medication is dispensed to be taken with breakfast. Bob's current weight and other statistics are added to previously collected data to determine health trends that may merit attention. When Bob leaves for work, MavHome reminds Bob remotely that he usually secures the home and has not done so today. Bob tells MavHome to finish this task and to water the lawn. Because there is a 60% chance of rain, the sprinklers are run a shorter time to lessen water

usage. When Bob arrives home, the hot tub is waiting for him. Bob has had a long day and falls asleep in the hot tub. After 40 minutes MavHome detects this lengthy soak as an anomaly and contacts Bob, who wakes up and moves on to bed [13]. MavHome's smart home capabilities are organized into a software architecture that seamlessly connects needed components while allowing improvements to be made to any of the supporting technologies. Figure 1 shows the architecture of a MavHome agent. The contributing technologies are separated into four cooperating layers [15]. The Decision layer selects actions for the agent to execute. The Information layer collects information and generates inferences useful for decision making. The communication layer routes information and requests between agents. The Physical layer contains the environment hardware including devices, transducers, and network equipment. The MavHome software components are connected using a CORBA interface. Because controlling an entire house is a very large and complex learning and reasoning problem, the problem is decomposed into reconfigurable subareas or tasks. Thus the Physical layer for one agent may in actuality represent another agent somewhere in the hierarchy, which is capable of executing the task selected by the requesting agent [15].

The database records the information in the Information layer, updates its learned concepts and predictions accordingly, and alerts the Decision layer of the presence of new data. During action execution, information flows top down. The Decision layer selects an action (e.g., run the sprinklers) and relates the decision to the Information layer. After updating the database, the Communication layer routes the action to the appropriate effectors to execute. If the effectors are actually another agent, the agent receives the command through its effectors and must decide upon the best method of executing the desired action. Specialized interface agents allow interaction with users and external resources such as the Internet [13][15]. Agents can communicate with each other using the hierarchical flow shown in Figure 1.

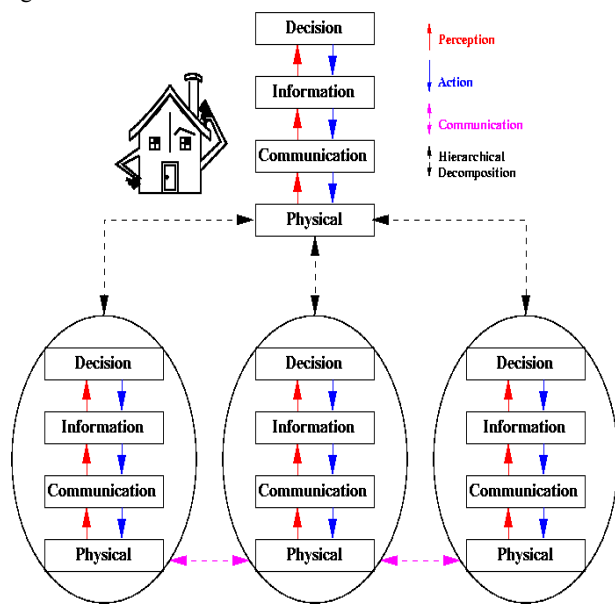


Figure 1: MavHome architecture [13]

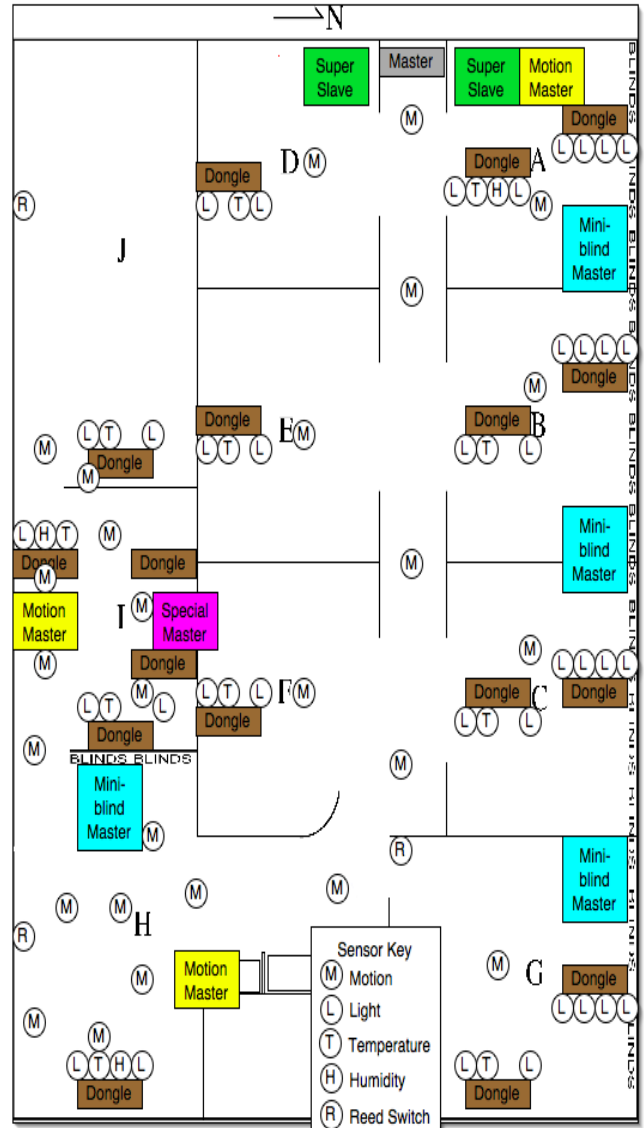


Figure 2: (a) MavHome argus sensor network [4]



Figure 2: (b) MavLab living room



Figure 2: (c) MavLab apartment kitchen

The Primary data collection system [4] consists of an array of motion sensors, which collect information using X10 devices and the in-house sensor network. Our dataset is collected for an inhabitant working in the MavLab (see Figure 2) and consists of two months of data. The lab consists of a presentation area, kitchen, student desks, and faculty room. There are over 100 sensors deployed in the MavLab that include light, temperature, humidity, and reed switches. In addition, we created a synthetic data generator to validate our approach. We developed a model of a user’s pattern which consists of a number of different activities involving several rooms and eight devices. For this paper we generated a data set containing about 4,000 actions representing two months of activities.

#### 4. TEMPORAL RELATIONS

Activities in a smart home include physical activities as well as instrumental activities. These may include walking, sitting on a couch, turning on a lamp, using the coffeemaker, and so forth. We see that these activities are not instantaneous, but have distinct start and end times. We also see that there are well-defined relationships between time intervals for different activities. These temporal relations can be represented using Allen’s temporal relations and can be used for knowledge and pattern discovery in day-to-day activities. These discoveries can be used for developing systems which can act as reminder assistants and also help detect anomalies and aid us in taking preventive measures [17].

Let us consider a scenario which involves a television, fan and a lamp being used in a smart home. We see that the inhabitant turns on the television and after some period of time turns on the fan. As time progresses, feeling cold, the fan is turned off and the individual continues watching the television. Later on, the television is turned off and the individual turns on the lamp to illuminate the room. We see that this scenario involved three activities each defined by interaction with a single device, namely a television, a fan and a lamp. Now we apply Allen’s logic to establish the temporal relations among the activities which occurred. The scenario is illustrated in figure 3. These activities can be represented as television “contains” fan and “meets” lamp. We can also represent these relationships as television “meets” lamp and fan “before” lamp.

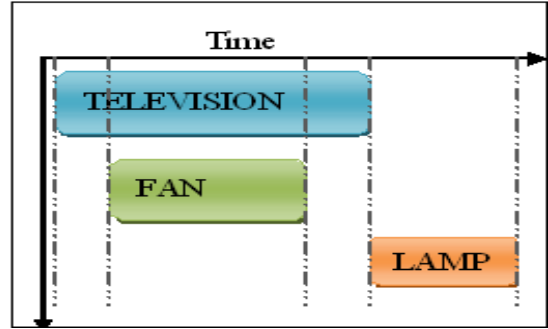


Figure 3. Illustration of temporal intervals

Table 1. Temporal relations representation.

Temporal Relations	Pictorial Representation	Interval constraints
X Before Y		$Start(X) < Start(Y);$ $End(X) < Start(Y)$
X After Y		$Start(X) > Start(Y);$ $End(Y) < Start(X)$
X During Y		$Start(X) > Start(Y);$ $End(X) < End(Y)$
X Contains Y		$Start(X) < Start(Y);$ $End(X) > End(Y)$
X Overlaps Y		$Start(X) < Start(Y);$ $Start(Y) < End(X);$ $End(X) < End(Y)$
X Overlapped-By Y		$Start(Y) < Start(X);$ $Start(X) < End(Y);$ $End(Y) < End(X)$
X Meets Y		$Start(Y) = End(X)$
X Met-by Y		$Start(X) = End(Y)$
X Starts Y		$Start(X) = Start(Y);$ $End(X) \neq End(Y)$
X started-by Y		$Start(Y) = Start(X);$ $End(X) \neq End(Y)$
X Finishes Y		$Start(X) \neq start(Y);$ $End(X) = End(Y)$
X Finished-by Y		$Start(X) \neq Start(Y);$ $End(X) = End(Y)$
X Equals Y		$Start(X) = Start(Y);$ $End(X) = End(Y)$

## 5. EXPERIMENTATION AND RESULTS

Sensor data from a smart environment can be represented and mined as sequences or as time series data. A sequence is an ordered set of events, frequently represented by a series of nominal symbols [5]. All the sequences are ordered on time scale and occur sequentially one after another. However, for some applications it is not only important to have a sequence of these events, but also a time when these events occur. A time series is a sequence of continuous real-value elements [5]. This kind of data is obtained from sensors which continuously monitor parameters such as motion, device activity, pressure, temperature, brightness, and so forth. Each time stamped data is characterized by specific properties. In table 2, we illustrate a sample of raw data collected from the sensor and also include the data as how it looks after it is processed and temporal intervals are identified.

**Table 2. Sample of sensor data across various stages of experimentation.**

Raw Sensor Data			
Timestamp	Sensor State	Sensor ID	
3/3/2003 11:18:00 AM	OFF	E16	
3/3/2003 11:23:00 AM	ON	G12	
3/3/2003 11:23:00 AM	ON	G11	
3/3/2003 11:24:00 AM	OFF	G12	
3/3/2003 11:24:00 AM	OFF	G11	
3/3/2003 11:24:00 AM	ON	G13	
3/3/2003 11:33:00 AM	ON	E16	
3/3/2003 11:34:00 AM	ON	D16	
3/3/2003 11:34:00 AM	OFF	E16	
Identify Time Intervals			
Date	Sensor ID	Start Time	End time.
03/02/2003	G11	01:44:00	01:48:00
03/02/2003	G19	02:57:00	01:48:00
03/02/2003	G13	04:06:00	01:48:00
03/02/2003	G19	04:43:00	01:48:00
03/02/2003	H9	06:04:00	06:05:00
03/03/2003	P1	10:55:00	17:28:00
03/03/2003	E16	11:18:00	11:34:00
03/03/2003	G12	11:23:00	11:24:00
Associated Temporal Relations			
Date time	Sensor ID	Temporal Relation	Sensor ID
3/3/2003 12:00:00 AM	G12	DURING	E16
3/3/2003 12:00:00 AM	E16	BEFORE	I14
3/2/2003 12:00:00 AM	G11	FINISHESBY	G11
4/2/2003 12:00:00 AM	J10	STARTSBY	J12

We validate our algorithm by applying it to our real and synthetic datasets. We train the model based on 59 days of data and test the model on one day of activities. We use the training set to form the frequent item sets and identify temporal relations shared between them. The temporal relations formed in these data sets show some interesting patterns and indicate relations that are of interest. The parameter settings pertaining to the training set data are given in Table 3. The parameter settings pertaining to the test set data are given in Table 4. Next we perform frequent itemset mining and identify the most frequent activities in the training dataset. Then we read these temporal relations into our prediction tool which dynamically calculates evidence for each possible event and

predicts the activity. We later observe the prediction and compare it with test data and report the results which include the accuracy and error in the observations

**Table 3. Parameters setting for training set.**

Datasets	Parameter Setting			
	No# Days	No # Events	No# Intervals Identified	Size of Data
Synthetic	59	8	1703	105KB
Real	59	17	1523	103KB

**Table 4. Parameters setting for test set.**

Datasets	Parameter Setting			
	No# Days	Number of Possible Events	Number of Intervals Identified	Size of Data
Synthetic	1	8	17	2KB
Real	1	17	9	1KB

The first step of the experiment is to process the raw data to find the temporal intervals. This is done using a simple tool which takes the timestamp of the event that occurred and based on the state (ON/OFF) forms the intervals. Later this data is passed through a temporal analyzer tool which identifies the temporal intervals based on the constraints formulated. The pseudo code for the temporal analyzer tool is described in Algorithm 1.

---

### Algorithm 1 Temporal Interval Analyzer

---

**Input:** data timestamp, event name and state

**Repeat**

**While** [Event && Event + 1 found]

Find paired "ON" or "OFF" events in data to determine temporal range.

Read next event and find temporal range.

Identify relation type between event pair from possible relation types (see Table 1).

Record relation type and related data.

Increment Event Pointer

**Loop Until** end of input.

---

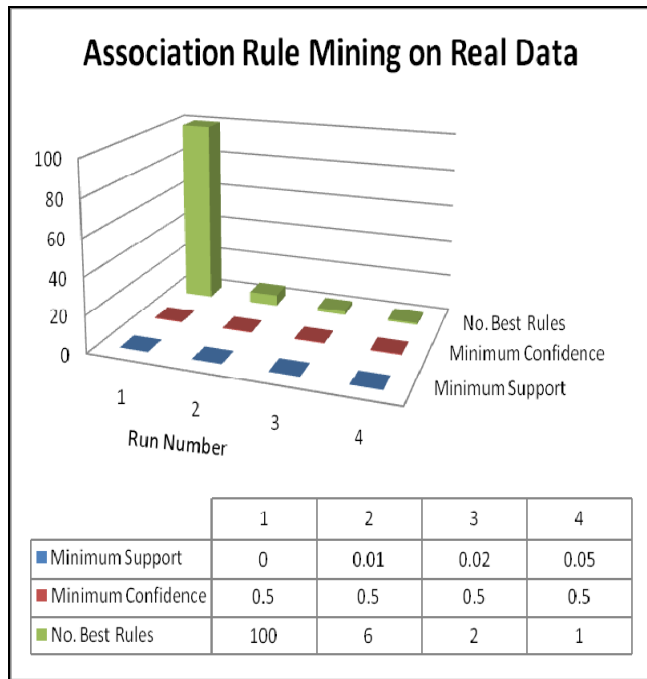
The next step of the process involves identification of the frequent activities, or events, which occur during the day and establishing temporal relations among them. To accomplish this task, we mine the data for frequent itemsets using the Apriori algorithm [17]. Next, we identify observed temporal relations between events in these frequent itemsets. We limit our approach to frequent activities, because the smart home data is so huge that there are many potential anomalies which are just factors of noise and thus make the datasets prohibitively large [17].

In the next step, we identify the association rules using weka [7] which can be used for prediction. The Weka implementation of an Apriori-type algorithm is used, which iteratively reduces the minimum support until it finds the required number of rules within a given minimum confidence. Table 5. summarizes the

parameters that were set and the number of rules generated with a given specified minimum confidence for the real dataset. Table 6 summarizes the same for the synthetic dataset.

**Table 5. Parameter settings and rules generated using Apriori-type algorithm in Weka for real dataset.**

Run#	Minimum Support	Minimum Confidence	No of Best Rules Found
1	0.00	0.5	100
2	0.01	0.5	006
3	0.02	0.5	002
4	0.05	0.5	001



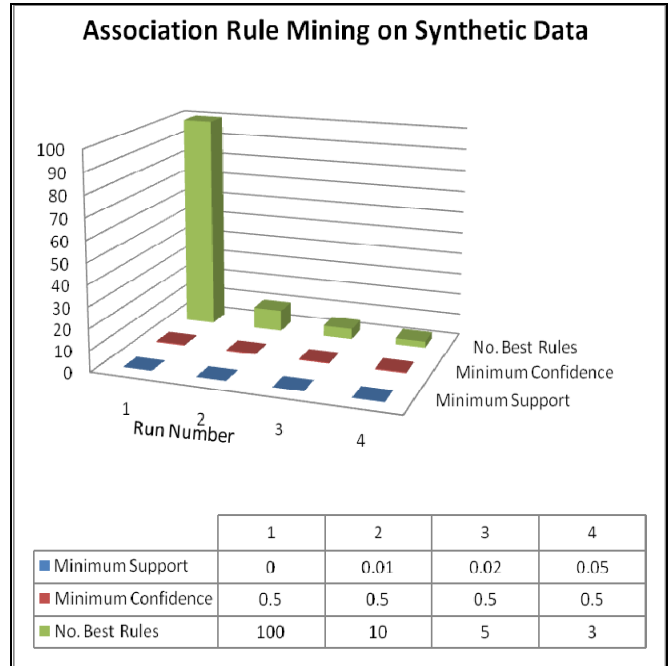
**Figure 4. Association rule mining in real datasets**

**Table 6. Parameter settings and rules generated using Apriori-type algorithm in Weka for synthetic dataset.**

Run#	Minimum Support	Minimum Confidence	No of Best Rules Found
1	0.00	0.5	100
2	0.01	0.5	010
3	0.02	0.5	005
4	0.05	0.5	003

We see that the figure 4 and figure 5 represent the observations from tables 5 and 6. We see that they represent various configurations which were used in Weka to find the best rules which can aid the prediction process. We observe here that when there is no minimum support we have many number of rules generated and as the support is increased we see that the number

of rules generated gets decreased. The sample illustration of the rules generated is given in table 6.



**Figure 5. Association rule mining in synthetic datasets.**

In Tables 5 and 6, the confidence level above 0.5 and support above 0.05 could not be used, as they could not result in any viable rules, due to the smaller datasets being used. As we see that the datasets are small, we use the top rules generated with a minimum confidence of 0.5 and a minimum support of 0.01.

**Table 7. Display of a sample of best rules generated.**

**Sample of best rules observed in real datasets:**

Activity=C11 Relation=CONTAINS 36 ==> Activity=A14 36  
 Activity=D15 Relation=FINISHES 32 ==> Activity=D9 32  
 Activity=D15 Relation=FINISHESBY 32 ==> Activity=D9 32  
 Activity=C14 Relation=DURING 18 ==> Activity=B9 18

The final step involves calculating the evidence of the event occurrence, which can be used for calculating the prediction. This step is to detect whether the particular event satisfies the temporal relations which can be used for prediction. These temporal relations which can be used for prediction are listed in Table 2. Let us look at an example where we have three frequent activities and are in the order of toaster , table lamp and radio, turning on and off in the morning. We see that the relation exhibited by them can be toaster “before” table lamp “finishes” radio. Now when the toaster and the radio occur without the table lamp, we can note that this is an anomaly in activity and we can use the same relation as when the toaster occurred and table lamp occurred then we can predict that the radio is going to occur in the near future before the table lamp is turned off. This method of prediction is based entirely on normative behavior as observed in the past and a strong rule is identified. As a result, the likelihood of prediction increases when there are strong repetitions of inhabitant patterns over time which are not anomalies. This method is a probability-

based model which involves calculating the evidence supporting the currently- occurring activity with respect to the previously- occurred activates and determining whether the current activity is an anomaly or not. This evidence calculation draws from the temporal relations which can be used for anomaly detection and which are listed in Table 8. Because of its robustness in the presence of noise, we decide to use this approach for our prediction.

**Table 8. Temporal relations which are used prediction.**

Temporal Relations	Usable for Prediction
Before	x
After	✓
During	✓
Contains	x
Overlaps	x
Overlapped-By	✓
Meets	x
Met-By	✓
Starts	✓
Started-By	✓
Finishes	✓
Finished-By	✓
Equals	✓

Let us consider two events X and Y and now focus our attention on events X and Y. We describe the prediction process for event X given information about an event Y that exhibits a temporal relationship with X in the following steps and has a strong rule identified by the association rule analysis. We see that this event Y is passed from the sequential predictor.

**Step A:** Learn temporal relations from the observed event history by analyzing the events and deriving the frequent itemsets. We also identify the most current activity (in our example, this is event Y). (Note that this tool can run simultaneously with a prediction component and take the output of the sequential prediction for processing).

**Step B:** Now we calculate the evidence supporting the occurrence of activity X. The formula to calculate the evidence using temporal relations is given by Equation (1). Note that equation is based on the observed frequency of the temporal relations, specifically those that influence the occurrence of event X.

$$P(X|Y) = \frac{|After(Y,X)| + |During(Y,X)| + |OverlappedBy(Y,X)| + |MetBy(Y,X)| + |Starts(Y,X)| + |StartedBy(Y,X)| + |Finishes(Y,X)| + |FinishedBy(Y,X)| + |Equals(Y,X)|}{|Y|} \quad (1)$$

$$Evidence_x = P(X) \quad (2)$$

The previous discussion showed how to calculate the likelihood of event X given the occurrence of one other event Y. Now consider the case where we want to combine evidence from multiple events that have a temporal relationship with X. In our

example we have observed the start of event A and the start of event B, and want to establish the likelihood of event X occurring. From Equation 1 we can calculate evidence

Now we have the evidence of B as:

$$P(B|A) = \frac{|After(B,A)| + |During(B,A)| + |OverlappedBy(B,A)| + |MetBy(B,A)| + |Starts(B,A)| + |StartedBy(B,A)| + |Finishes(B,A)| + |FinishedBy(B,A)| + |Equals(B,A)|}{|A|} \quad (3)$$

Similarly when we have the events occurred as follows: **A B X**

Now the evidence of B is calculated as follows:

$$P(X|AUB) = \frac{P(X \cap (A \cup B))}{P(A \cup B)}$$

$$= \frac{P(X \cap A) \cup P(X \cap B)}{P(A) + P(B) - P(A \cap B)}$$

[Association Rule]

$$= \frac{P(X|A).P(A) + P(X|B).P(B)}{P(A) + P(B) - P(A \cap B)}$$

[Multiplication Rule] (4)

And we see that, we can use the previous calculated evidence for calculating newer evidence, based on the equation (4). We see that in this equation (4), uses Association rule and Multiplicative rules to arrive at the final formula which includes previous computed evidences of occurred events. And we use them to calculate the evidence of the most recent occurred event. In this way we compute the evidence of the occurred events and applies to the entire series of events that occurred.

**Step C:** Now we finally calculate the prediction of X by the equation (5) given below.

$$Prediction_x = P(X) \quad (5)$$

Notice that if the event has a probability approaching 1 and has occurred, this is considered most likely to occur. On the other hand, if the probability of the event we just observed is close to 0, then this is an unusual event and should not be considered for prediction. The point at which these predictions are to be considered surprising enough to be reported is based somewhat on the data itself. If the probability of an event is based on the occurrence of other events which themselves rarely occur, then the evidence supporting the occurrence of the event is not as strong. In this case, if the event has a low probability yet does occur, it should be considered less anomalous than if the supporting evidence itself appears with great frequency. Consistent with this theory, we calculate the mean and standard deviation of event frequencies over the set of events in the inhabitant's action history. Events are reported as predicted (or, conversely, the absence of an event) if it does occur and its prediction value is greater than the mean + 2 standard deviations.

This indicates an improvement of prediction in a single day of the inhabitant in smart environment compared to a simple sequential prediction technique employed earlier.. The main reason for a significant error rate is the smaller amount of data used. As we have larger datasets we see that the performance of the temporal relations enhanced prediction would also improve drastically over time.

The final step involves enhancing our ALZ predictor [9] by adding these temporal relations to the input data and comparing the performance with and without the rules. This sequential predictor employs incremental parsing and uses Markov models for prediction. We notice that many situations demand that the prediction algorithm be capable of analyzing information and

delivering in real time. We currently plan to run real time analysis over large sets of data in the near future. The algorithm 2 specifies the temporal enhancement.

**Algorithm 2** Temporal Rules Enhanced prediction.

**Input:** Output of ActiveLeZi Predictor a, Best Rules r, Temporal Dataset

**Repeat**

**If** a! = null

**Repeat**

      Set r1 to the first event in the first best rule

**If** (r1 == a) **Then**

**If** (Relation! = "After") **Then**

          Calculate evidence (use Equation 1)

**If** evidence > (Mean + 2 Std. Dev.) is noted

**Then**

            Make event related to r1 in the best rule as next predictor output;

**Else**

            \*Get next predicted event and look for there temporal relation in the temporal relations database based on the frequency, calculate evidence. store temporarily in buffer

**If** again the relation is after **Then** goto \*

**Until** no more "After" relations are found

            Calculate evidence

**If** evidence > (Mean + 2 Std. Dev.)

**Then** predict;

**Else**

              Calculate evidence and if evidence > (Mean + 2 Std. Dev.) **Then** predict this event based on the relation;

**End if.**

**Until** end of rules.

**End if.**

**Loop until** End of Input.

**Table 9. Comparing ActiveLeZi based prediction with and without temporal rules in real datasets.**

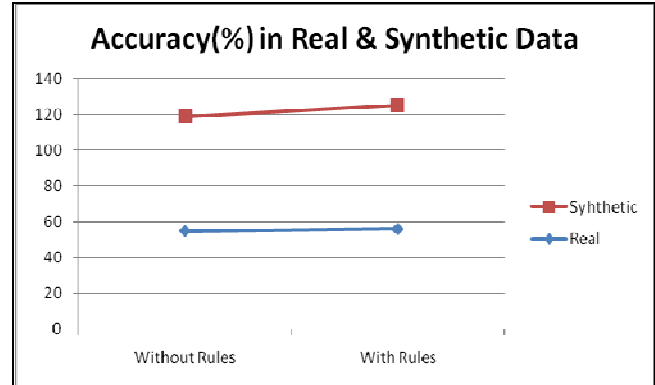
Datasets	Percentage Accuracy	Percentage Error
Real (Without Rules)	55	45
Real (With Rules)	56	44

**Table 10. Comparing ActiveLeZi based prediction with and without temporal rules in Synthetic datasets.**

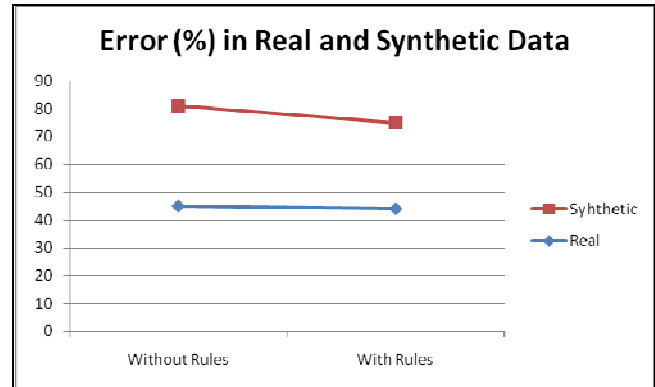
Datasets	Percentage Accuracy	Percentage Error
Synthetic (Without Rules)	64	36
Synthetic (With Rules)	69	31

We see that the Table 9 and Table 10 present us the results observed in the prediction experiment and compare the percentage accuracy with the percentage error noted in the results. We see that the illustration of the observed accuracy and error in the real and synthetic datasets is given the figure 6. and figure 7. respectively. We can note that in figure 6 we see that the accuracy

increases with temporal rules and in figure 7 we note that the error curve decreases.



**Figure 6. Percentage accuracy in real and synthetic datasets.**



**Figure 7. Percentage error in real and synthetic datasets.**

Table 9. and Table 10., above summarizes the observed accuracy of the prediction performance on real and synthetic datasets. We see that there was 1.86% prediction performance improvement in the real data and 7.81% improvement in the synthetic data. This indicates an improvement of prediction in a single day of the inhabitant in smart environment. The main reason for a significant error rate is the smaller amount of data used. As we have larger datasets we see that the performance of the temporal relations enhanced prediction would also improve drastically over time. Another reason for lesser accuracy is because of the used test set as it need not be a typical day or could be one where the inhabitant had very less presence in the smart environment. Overall we see a unique application of temporal relations based mining being applied.

Now the future work here extends the concept of temporal relations and apply it directly to the ActiveLeZi prediction component [9] by enhancing its probability calculating component to incorporates temporal probability which would help the prediction to perform better. This can be illustrated by the equation 6 below. In the equation 6, C & P represent child and parent respectively in the binary tree formed by the ALZ predictor for prediction.

$$\text{Prediction}_C := P(C|P) = P(C|P)_{\text{SEQ}} + P(C|P)_{\text{TEM/Global}} - (\alpha * P(C|P)_{\text{TEM}})$$

$$\text{Where } \alpha = |C_{\text{PHRASE}}| / |C_{\text{GLOBAL}}|. \tag{6}$$



## 6. DATA VISUALIZATION

Another notable problem which is observed is how can we visualize the temporal intervals identified. An interesting question would be say, how the inhabitant could compare his average time to cook in a smart environment to average time with other activity such as turning on the oven. This problem is handled here with the help of a smart interval visualization tool being developed. In many application domains temporal data queries are fundamental to understand and reason over time and space. However it still would be a difficult challenge to describe and formalize time using physical characteristics and properties. Thus, we see that we can only establish a temporal relation from the observations. In this paper, we also present an interface which helps us visualize activity intervals identified in smart home datasets [17]. We can also use this tool as a media for looking for patterns and also acts as a user-friendly data visualization and manipulation tool. The end goal of this tool is to enable the resident to visualize interesting patterns over time and help him improve his lifestyle at home by identifying activities which hinder the growth and health and also identify activities which help him to have an improved lifestyle.

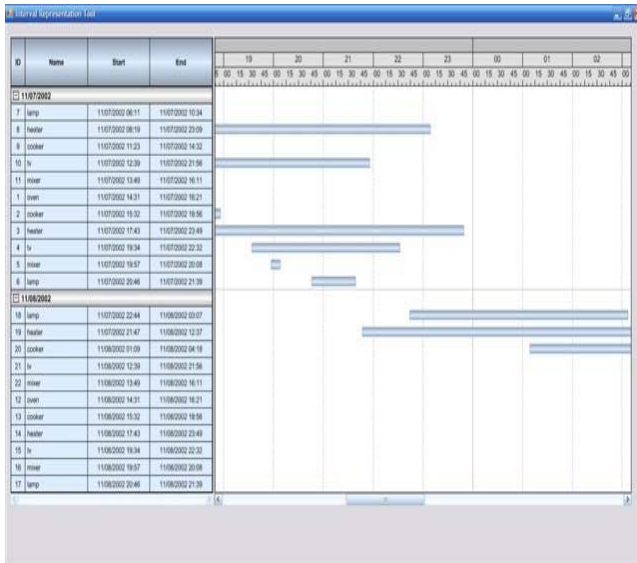


Figure 8. Interval Visualization Tool

This tool gives the inhabitant complete control of visualization with scrolling and also enabled to look for patterns. This tool was developed using C#.NET with Visual studio2005 IDE and also uses VARCHART Xgantt plotting library (a 3<sup>rd</sup> party library) for visualization in the form of Gantt charts which help represent the activity intervals over a time period [10] [12]. A screenshot of this tool is displayed above in the figure 8. In this illustration, we can visualize the various activities with their event IDs, the name of the activity/sensor, its start time and end time and the corresponding time interval.

## 7. FUTURE WORK

While making sense of sensor data can be challenging for smart environment applications, the problem is made even more complex when the environment houses more than one resident. To aid the capabilities of our temporal data mining, and to reveal the complexities of multi-inhabitant spaces, an entity discovery

tool is needed. Enriching the raw data set provided by the tagging multiple inhabitants would make Knowledge Discovery more interesting to work with for determining features of the data during the mining stages. In this case, we are enriching the data with information about entities moving within the space. This comes in the form of an entity (in this case, resident) identification number that is attached to each event, matching events to entities. As an entity traverses the space they trigger events, but historically these events have primarily been attributed to individuals using some form of 3<sup>rd</sup> party tracking.

## 8. CONCLUSION

Time is an important aspect of all real world phenomena. Data mining is concerned with analyzing large volumes of data to automatically discover interesting regularities or relationships which in turn lead to a better understanding of the underlying processes [6] [16]. Within this field, temporal data mining focuses on such analysis in the case of ordered data streams with temporal interdependencies. Temporal rule based knowledge discovery is a new area in smart home research. We notice that the use of temporal relations provides us a unique new approach for prediction. The current approach is currently experimented on small datasets, but we will next validate the performance of our algorithm on larger datasets. Some future directions of this work also includes the expansion of the temporal relations by including more temporal relations, such as until, since, next, and so forth, to create a richer collection of useful temporal relations.

## 9. ACKNOWLEDGMENTS

This work is supported by NSF grant IIS-0121297.

## 10. REFERENCES

- [1] A. Helal, W. Mann, H. El-Zabadani, J. King, Y. Kaddoura, and E. Jansen. The Gatortech smart house: A programmable pervasive space. *IEEE computer*, 38(3):50-60, 2005.
- [2] Björn Gottfried, Hans W. Guesgen, and Sebastian Hübner, Spatiotemporal Reasoning for Smart Homes. *Designing Smart Homes, Lecture Notes in Computer Science, Springer Berlin / Heidelberg*, Pg 16-34, Volume 4008/2006, July 2006.
- [3] F. Doctor, H. Hagra, and V. Callaghan. A fuzzy embedded agent-based approach for realizing ambient intelligence in intelligent inhabitant environment. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 35(1): 55-65, 2005.
- [4] G. Michael Youngblood, Lawrence B. Holder, and Diane J. Cook. Managing Adaptive Versatile Environments. *Proceedings of the IEEE International Conference on Pervasive Computing and Communications*, 2005.
- [5] Guralnik V. and Srivastava J. Event detection from time series data. *Proceedings of fifth ACM SIGKDD international conference on knowledge discovery and data mining*, California, US, pp 33-42, 1999.
- [6] Hand D, Mannila H, and Smyth P. Principles of Data Mining: A Bradford Book. *The MIT Press*, Cambridge, Massachusetts; London, England, 2001.
- [7] Ian H. Witten, Eibe Frank. 2005. Data Mining: Practical Machine Learning Tools and Techniques, *2nd Edition. Morgan Kaufmann*, San Francisco.

- [8] James F. Allen, and George Ferguson, Actions and Events in Interval Temporal Logic, *Technical Report 521*, July 1994.
- [9] K. Gopalratnam & D. J. Cook (2004). Active LeZi: An Incremental Parsing Algorithm for Sequential Prediction. *International Journal of Artificial Intelligence Tools*. 14(1-2):917-930.
- [10] Microsoft Corporation. Visual Studio. *Microsoft Publishing*. Retrieved May 28, 2007. <http://msdn.microsoft.com/vstudio/>
- [11] Mörchen, F. A better tool than Allen's relations for expressing temporal knowledge in interval data, *In Proceedings the Twelveth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Philadelphia, PA, USA, 2006.
- [12] Netronic. VARCHART XGantt. *NETRONIC Software*. Retrieved April 2007. [http://www.netronic.com/english/index\\_prod.html?set\\_x\\_gantt.html](http://www.netronic.com/english/index_prod.html?set_x_gantt.html)
- [13] S. Das and D. J. Cook. Health Monitoring in an Agent-Based Smart Home. *Proceedings of the International Conference on Smart Homes and Health Telematics (ICOST)*, Singapore, September, 2004.
- [14] S. Das and D. J. Cook, Smart Home Environments: A Paradigm Based on Learning and Prediction. *Wireless Mobile and Sensor Networks*, Wiley, 2004.
- [15] S.K. Das, and D. J. Cook. Smart home environments: A paradigm based on learning and prediction. In *Wireless Mobile and Sensor Networks: Technology, Applications and Future Directions*. Wiley, 2005.
- [16] W. Frawley, G. Paitesky-Shapiro, C. Matheus, Knowledge Discovery in Databases: An overview, *AI Magazine*, Vol 13 n. 3 (57-70). 1992.
- [17] Vikramaditya Jakkula, and Diane J. Cook, Learning temporal relations in smart home data, *Proceedings of the second International Conference on Technology and Aging*, Canada, June 2007.
- [18] A Dekhtyar et al, Probabilistic temporal databases, I: algebra, *ACM Transactions on Database Systems (TODS)*, Volume 26 , Issue 1 (March 2001).
- [19] V. Ryabov et al, Probabilistic Reasoning about Uncertain Relations between Temporal Points, *Eighth International Symposium on Temporal Representation and Reasoning (TIME'01)*, 2001.
- [20] Hornsby, K.H. and Worboys, M.F., Event-oriented approaches in geographic information science, (A special issue of Spatial Cognition and Computation 4(1)), Lawrence Erlbaum, Mahwah, NJ, ISBN: 0-8058-9531-0, 2004.