Temporal Motif Mining Approaches for Smart Homes

Huijuan Shao[1,2], Yaowei Li[3], Fei Li[2], Erin Griffiths [4], Kamin Whitehouse[4], Naren Ramakrishnan[1,2]
[1]Discovery Analytics Center, Virginia Tech, Blacksburg, VA 24061
[2]Department of Computer Science, Virginia Tech, Blacksburg, VA 24061
[3] Game source, McLean, VA 22101
[4]Department of Computer Science, University of Virginia, VA 22904

ABSTRACT

Conserving energy and optimizing its use has been a long standing challenge. Apart from the monetary benefits associated with tackling these problems, saving energy has significant positive environmental impact. For instance, it would be useful to automatically adjust the HVAC of residential buildings based on occupancy. In this work, we mine people's energy activity profile to predict the occupancy of residential buildings. I propose a novel hybrid method, which uses episode mining for target event detection and a mixture of episode-generating HMM (EGH), combined with the standard kNN approaches and demonstrate how this hybrid approach always yields the best results.

KEYWORDS

occupancy prediction, motif mining, hidden Markov model

ACM Reference format:
Huijuan Shao[1,2], Yaowei Li[3], Fei Li[2], Erin Griffiths [4], Kamin Whitehouse[4], Naren Ramakrishnan[1,2]
[1]Discovery Analytics Center, Virginia Tech, Blacksburg, VA 24061
[2]Department of Computer Science, Virginia Tech, Blacksburg, VA 24061
[3] Game source, McLean, VA 22101
[4]Department of Computer Science, University of Virginia, VA 22904
. 2016. Temporal Motif Mining Approaches for Smart Homes. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 8 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Modeling activity of daily life (ADL) has become a burgeoning research topic, since people demand a comfortable life at home at a lower cost. Since heating and cooling spaces consumes $\sim$53% of the total electrical usage by heating and cooling spaces of an average household, automating the operation of HVAC devices to save energy is important. One of the crucial components required to achieve this goal is to model and predict the occupancy of a home. Supervised learning approaches on the analysis of indoor temperature [5], smart phones' GPS data [6], electricity consumption [3] and sensor data by tracking indoor activities [1, 13] are effective ways to approach this prediction problem. Prediction of occupancy using sensor data has been broadly researched. By capturing daily activities, like room occupancy of the house, usage of electrical devices, and usage of water systems using sensors, researchers have modeled occupancy [2, 3, 9] and used these results to automate the control of the HVAC system.

Although the supervised learning kNN [13], neural network [9] and Markov model [3] are effective, the detailed household activities represented as a time series are not fully utilized. Daily activities such as waking up, cooking, washing, and commuting to work/school and back have different patterns based on the day. For instance, the schedule on a working day is significantly different from that of a weekend or a holiday. Thus, this scenario leads itself to episode mining analysis, which can be used to predict household occupancy. Using this strategy of episode mining for occupancy prediction has three advantages. First, episode mining, a temporal mining approach, mines according to the time distribution for each type of activity. Second, it builds the activity scenario and connects the episode with a probabilistic hidden Markov model (HMM). Unlike previous models, the time and order of each kind of activity are fully utilized. Third, the algorithm predicts according to the scenario-based probabilistic model episode generative HMM (EGH). The prediction accuracy is better than the existing models.

This work's contributions can be highlighted in the form of the three questions below.

(1) How can we mine for meaningful scenarios? Episode mining can mine many frequent episodes, but not all the episodes are useful for occupancy prediction. By narrowing the episodes according to the start state, end time, event dwelling time and the gap between two activities, we can interpret these episodes and provide insight as to which episodes are informative.
(2) How can we predict the occupancy more accurately? Our dataset comprises detailed information of the various activities of a household tracked as a time series on a daily basis. Thus our episodes have rich detailed information based on occupancy and unoccupancy of the household. Since we are mining episodes from this data, the accuracy of occupancy prediction improves significantly.
(3) Can it help save electric usage at home? The prediction occurs at least 15 minutes ahead of a person leaving or coming back. By connecting this prediction result to an automatic HVAC controlling system, the HVAC can be turned on or off ahead of the occupancy change. Since the HVAC does not work during occupancy, this saves electric usage.

Conference'17, Washington, DC, USA

DOI: 10.1145/nnnnnnn.nnnnnnn

2 RELATED WORK

Accurately predicting whether a home is occupied is a difficult task. People in the same home have different daily schedules; some go to work and others stay at home for a period of time. A great deal of research has been done to track the activities of people to infer the home occupancy. Researchers have made efforts to collect data by sensors, smart phones, the calendar, and weather information. Most of the approaches that model and predict occupancy primarily use sensor data to detect conditions such as room occupancy, use of electrical appliances, water usage, etc. Several supervised learning approaches, such as kNN, neural networks, rule-based models, and Markov chain models have been used to model and predict building occupancy [1–3, 9, 13]. Using the kNN supervised learning algorithm and monitoring sensor data for a portion of the day, Scott et al. predict an entire day's occupancy in [13]. A neural network approach using a binary time series based on occupancy/unoccupancy along with exogenous input network (NARX) is proposed in [9]. Mahmoud et al. tackle the problem by presenting a non-linear autoregressive model with an exogenous input (NARX) network. Several Markov chain models, like the blended Markov chain, closest distance Markov chain, and moving-window Markov chains are presented in [3]. A mixture of multi-lag Markov chains was used to predict the occupancy of single-person offices [10]. In that work, the authors also compare their model with the Input Output Hidden Markov Model, First Order Markov Chain and the NARX neural network.

A recent survey [5] compares major occupancy predictions algorithms against the LDCC dataset [4], which was collected by GPS and other sensors. It shows that time-based presence probability [7] performs slightly better than the preheat kNN approach [13]. Since the preheat kNN approach [13] is more widely applicable, in that it can be used against both GPS and sensor datasets, we set it as a baseline method for comparison.

3 PROBLEM FORMULATION

Given M time series, each time series $X^{(m)} = X_1^{(m)}, ..., X_t^{(m)}, ..., X_T^{(m)}$ represents a sequence of room occupancy of person m inside a home over K days, where $X_t \in s$ denotes that X belongs to a finite room set s at the sequence number of t, and $m \in \{1, ..., M\}$. Let Z denote that the home is unoccupied and $Z \in s$. We predict whether person m stays at home the rest of a day from time T, i.e. during $T+1, T+2, ..., \Delta T$,

$$\hat{Y}^{(m)} = \hat{Y}_{T+1}^{(m)}, ..., \hat{Y}_{T+\Delta T}^{(m)} \tag{1}$$

where $Y_{T+\Delta t}^{(m)} = Z$ if person m does not stay at home at time $T + \Delta t$; otherwise, $Y_{T+\Delta t}^{(m)} \neq Z$. If any person m stays at home $Y_{T+\Delta t}^{(m)} \neq Z$, then this house is occupied $Y_{T+\Delta t} \neq Z$.

4 TEMPORAL MINING MIXTURE MODEL

We use a three-pronged approach to tackle the problem of mining and predicting unoccupancy, as shown in Figure 1. Given indoor activities time series of a person over a period of time, first, we use an episode mining algorithm to discover frequent episodes from the past days' data. Then we connect each episode with an EGH and build a mixture EGH model. Based on the mixture model, we predict when each person will leave and come back the house. If all people leave, then the house is unoccupied.

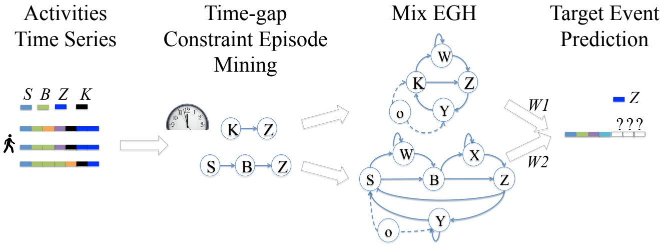
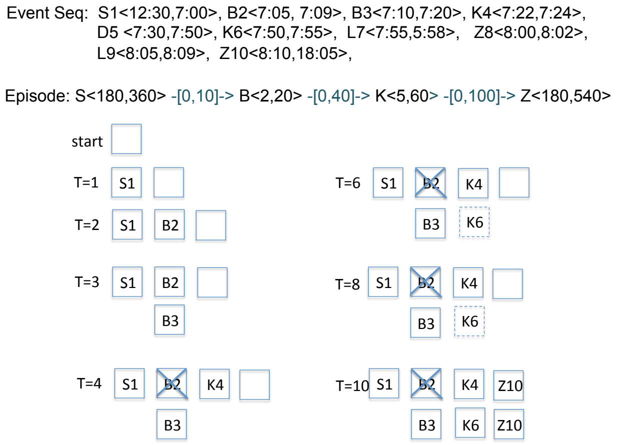


Figure 1: Occupancy Prediction Framework.

Episode An episode is a collection of ordered events. Here an episode refers to an ordered events which are highly relevant to the occupancy status inside a building. For instance, we represent 'S' as sleep, 'K' as kitchen, and 'Z' as going out. If an episode $S \rightarrow K \rightarrow Z$ is found, the story is described as a person getting up, going to the kitchen for breakfast, and then leaving the house. An episode α is composed of a series of ordered events $\alpha = \langle X_1, .., X_t, ...X_T \rangle$, where X_t denotes that X occurs at a sequence of t. The event X_t may be the point event or dwelling event. The dwelling event has a start time $X.start$ and end time $X.end$. In this paper X denotes a dwelling event and represents which room a person stays inside a building, i.e., this building is *occupied*. Since Z denotes a room is unoccupied, $Z.start$ is the point at which a person or all people inside the building leave, and $Z.end$ is the point at which a person or all people come back.

4.1 Time-gap Constraint Episode Mining

Episode Mining Episode mining has been studied in previous research [11]. It uses a non-overlap mining approach to find the frequent episodes. Episode mining has been applied to energy disaggregation to help conserve energy in buildings [14] in sustainability research. In contrast with previous research, the events in this application of occupancy prediction dwell at an event for a period of time. As a result, we extend the above two episode mining algorithms [8, 12] and enforce more constraints. One change is to adopt right alignment for the first element in the episode mining. The second modification is to add time constraints and apply gap duration constraints between two consecutive events inside an episode. Figure 2 shows a time-gap constraint episode mining example. Let us assume we have a frequent episode $S \rightarrow B \rightarrow K \rightarrow Z$. We then add the time constraints to each event $\{S, B, K, Z\}$. The dwelling duration of S is 3 to 6 hours, of B is 2 to 20 minutes, of K is 5 to 60 minutes, and of Z is 3-9 hours. In addition, we set gap duration between any two consecutive events. The gap duration of SB is calculated as $\Delta SB = B.start - S.end$. We set the maximal gap time between SB, BK, and KZ as 10 minutes, 40 minutes and 100 minutes; the minimal gap time is 0. Then we have a stream composed of the sequence of dwelling events "Event seq," as shown in Figure 2. The time-gap constraint episode mining process to discover a frequent episode uses the following method. Let the $node$ structure denotes each element in any episode, as depicted as a square box in Figure 2. Let $waits$ refer to a structure which pairs with an episode



Event Seq: S1<12:30,7:00>, B2<7:05, 7:09>, B3<7:10,7:20>, K4<7:22,7:24>, D5 <7:30,7:50>, K6<7:50,7:55>, L7<7:55,5:58>, Z8<8:00,8:02>, L9<8:05,8:09>, Z10<8:10,18:05>,

Episode: S<180,360> -[0,10]-> B<2,20> -[0,40]-> K<5,60> -[0,100]-> Z<180,540>

Figure 2: Time-gap constraint episode mining example.

and has the same length of this episode. Initially, a $waits$ structure related to episode $S \to B \to K \to Z$ is created. A $node$ structure related to S is created and it waits for the first element of the episode $S\langle 180, 360\rangle$. When $T = 1$, the duration of S_1 is checked. Since S_1 is in the range of $3 - 6$ hours, S_1 passes and is put into the node structure $node$ related to S. Next a new $node$ structure is created to wait for $B\langle 2, 20\rangle$. When $T = 2$ and $T = 3$, both B_2 and B_3 are qualified in terms of the time constraints and the gap constraints; e.g., the gap between S and B ΔSB should be between 0 and 10 minutes. These two nodes B_2 and B_3 are then input into the $waits$ structure. At the same time, a new $node$ structure is created for $K\langle 5, 60\rangle$. When $T = 4$, the gap between $\langle B_3, K_4\rangle$ is satisfied with the distance condition between B and K 0-40 minutes. However, the gap between $\langle B_2, K_4\rangle$ is longer than the constraint gap. Therefore, B_2 is canceled out. Now a new Z waits for the symbol $Z\langle 180, 540\rangle$. When $T = 6$, the gap from B_2 to K_6 is too large. Therefore, K_6 is not added into the $node$ K structure in $waits$. When $T = 8$, the duration of Z_8 does not qualify for the condition of between 3-9 hours, so Z_8 is not added. When $T = 10$, the duration of Z_{10} meets the requirement of between 3-9 hours and its distance to $K4$ meet the requirement of $\Delta KZ \in [0, 100]$ minutes. Thus Z_{10} is added into the $node$ Z structure in $waits$. Therefore, complete mining of an episode is complete, and we have mined two instances here; $S_1B_3K_4Z_{10}$ and $S_1B_3K_6Z_{10}$.

4.2 Mixture EGH

Episode Generating HMM Episode generative HMM (EGH) model is a type of HMM model which connects with frequent episode, and the more frequent an episode inside a sequence, the likelihood of the state sequence including this episode is larger [8]. The uniqueness of the EGH is that the transition matrix and emission matrix is only decided by a noise parameter η. The noise parameter η of frequent episode α is calculated as $\eta = \frac{T - Nf_\alpha}{T}$, where T is the training data stream length, α is the frequent episode, N is the length of frequent episode α, f_α is the frequency over the time T.

In the mixture EGH model shown in Figure 3, the transition matrix of an EGH is given as an example. Let us assume we have a N-node frequent episode $S \to B \to Z$, where $N = 3$. We define 2N number of hidden states; N for episode states, and N for noise states. The noise states are $\{W, X, Y\}$. An episode state transfers to another episode state at the probability of $1 - \eta$. An episode state transfers to a noise state at a probability of η. A noise state transfers to another noise state at a probability of $1 - \eta$. To calculate the emission matrix, first we let M denote the total number of symbols in the event stream. For any hidden states in the episode, M has a delta function emission. Whenever it is visited (right alignment of the first element in the episode, left alignment for the left elements in the episode), it will generate the same observation symbol. For any noise hidden states, it emits any of the symbols from the M observation symbols with a uniform distribution at probability $\frac{1}{M}$.

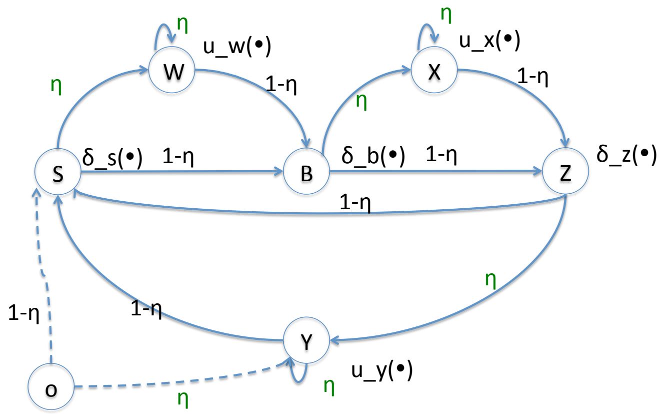


Figure 3: States Transition of Episode Generating HMM (EGH).

Theorem 1 from [8] is crucial. The theorems in [8] prove that the more frequent an episode is inside a sequence, the greater the likelihood of the state sequence including this episode. The proof for this theorem is explained in detail in [8].

THEOREM 1. *[8] Let $D_Z = X_1, ..., X_K$ is the given sequence data, ε is the symbol set, and the size of these symbols is M. Given two frequent N-node episodes α and β with frequency f_α and f_β. Their corresponding EGH is Λ_α and Λ_β. The most likely state sequence for episode α and β are q^*_α and q^*_β. The noise parameters of these two EGH are η_α and η_β. Assume both of these noise parameters are less than $\frac{M}{M+1}$, we have (1) if $f_\alpha > f_\beta$, then $P(D_Z, q^*_\alpha|\Lambda) > P(D_Z, q^*_\beta|\Lambda)$ (2) if $P(D_Z, q^*_\alpha|\Lambda) > P(D_Z, q^*_\beta|\Lambda)$, $f_\alpha > f_\beta$*

Mixture Model The mixture EGH model is fully discussed in previous work [8]. This model gives different weights to each EGH to predict a target event. We can assume we obtain whether an episode occurs on a certain day. Let $D_Z = \{X_1, ..., X_K\}$ denote the K days data set. $F = \{\alpha_1, ...\alpha_J\}$ denote the frequent episodes in the dataset D_Z. An EGH Λ_{α_j} is associated with frequent episode α_j. Λ_Z denotes a mixture EGH model. The likelihood of D_Z under the mixture model is written as Equation (2).

$$Pr(\Lambda|Z) = \prod_{i=1}^{K} P[X_i|\Lambda_Z] \quad (2)$$
$$= \prod_{i=1}^{K} (\sum_{j=1}^{J} \theta_j P[X_i|\Lambda_{\alpha_j}]) \quad (3)$$

where θ_j is the mixture coefficient of Λ_{α_j} and it subjects to $\sum_{j=0}^{J} \theta_j = 1$

The parts inside Equation (2) are additive; the coefficients θ are computed by EM algorithm. During the initialization part of the EM algorithm, the episode frequency over the times series T over K days is calculated. Specific frequent episodes ended with target event 'Z' are selected. Optionally, we could add special constraints on episodes, starting with certain event type 'S'. In the expectation step, one key part is the likelihood value of each episode α_j in time series X_i. The likelihood value is computed as Equation (4); Then Bayes rules is applied to compute the new coefficient θ_{new}.

$$Pr(X_i|\Lambda_{\alpha_j}) = (\frac{\eta_{\alpha_j}}{M})^{|X_i|} (\frac{1-\eta_{\alpha_j}}{\eta_{\alpha_j}/M})^{|\alpha_j| f_{\alpha_j}(X_i)} \quad (4)$$

In the maximization step, we update the objective value based on Equation (2) until it converges, i.e., until the difference of two consecutive objective values is smaller than a threshold.

4.3 Predict When the Target Event Occurs

Target event prediction is studied in [8], but only insofar as it predicts whether a target event will occur, rather than when the target event will happen. Our occupancy prediction algorithm enriches the previous event prediction algorithm by breaking into three subproblems; whether the target event un-occupancy Z will appear, when the target event Z starts, and when the target event Z ends.

Since the solution to the first sub-problem is similar to previous work, this sub-section emphasizes on the last two sub-problems. After obtaining the result of the first sub-problems, we assume we already know that the target event Z will surely happen, and so we need to predict when the person leaves or comes back. The leaving time corresponds to the start time of dwelling event $Z.start$ and the returning time refers to the end time of dwelling event $Z.end$.

After running episode mining and mixing the EGH model, we have obtained all the frequent episodes $F = \langle \alpha_1, ..., \alpha_J \rangle$, the corresponding EGH $\Lambda_{\alpha_j}, j = 1...J$ with noise parameter η_j, and the mixture models Λ_Z with coefficients θ_j. We use the coefficient of these mixture models to predict the leave time and return time of target event Z. Each day is cut into three phases: 1) Before a person gets up; 2) After the person gets up but before the person goes out; and 3) After the person goes out but before they come back.

(1) Usually before a person gets up, there is only one frequent episode named 'SZ'. The start time and end time of Z depends on 'S'. Therefore $Z.start$ and $Z.end$ are calculated by the probability density function of going out and coming back time in the past days.

(2) After the person gets up, if he/she has a lot of activities at home, there are several frequent episodes that are mined before the person leaves home. If there are several frequent episodes ending with Z, the leave time and return time of each episode is checked to determine whether they are in a range of probability density function (PDF) value in the past. If yes, the mean value of these episodes are recorded. Since each episode generates an EGH, the mixture EGH model computes a weight for each EGH as coefficients. The leave time $Z.start$ and back time $Z.back$ are the weighted mean leave time and back time of these frequent episodes.

(3) After a person leaves home, we already know when the person leaves home $Z.start$. If the person has come back, nothing needs to be predicted. If the person has not come back, the return time $Z.end$ is the weighted historical return time of mined frequent episodes, viz. the probability density function of backing time based on the time-constraint going out time.

5 EXPERIMENT RESULTS

We have conducted experiments on three datasets, where each dataset is obtained by monitoring 24-hour activities of two adults in a house via RFID. All these activities occur in twelve rooms; the basement, bathroom, bedroom, dining room, hallway, kitchen, living room, mudroom, nursery, outside-front, outside-back, and upstairs. The dataset comprises events in the form of timestamped room occupancy data points. For instance, an event can correspond to person 1 being in the kitchen at 7:00 am. The summary of these three datasets is shown in Table **??**. We define *unoccupancy* of a person as one of these conditions: the person leaves the *outside-front* or *outside-back* for more than 30 minutes; the person stays in the living room or dining room for more than 9 hours without any other activities; or the gap between any two events is more than 30 minutes. Since our research goal is to automate the turning on and off of the HVAC system at least 30 minutes before occupancy, the first and third constraints are in place. We are only interested in events where the *unocupancy* period is for an extended duration (> 30 minutes). The second constraint comes from our observation that if a person stays in one room for more than 9 hours without moving to other rooms, this usually means that the person has gone out but left the RFID equipment at home. Furthermore, we delete events with a duration less than 2 minutes since these correspond to the individual walking back and forth across rooms and generally do not contribute to meaningful episodes. We conduct four types of experiments to compare four approaches; kNN, mixture EGH, PDF, and support vector regression (SVR). For each dataset, we use $2/3$ of the data for training and the remaining $1/3$ of the data as a test. Following the approach in [13], we organize one day's data into 96 15-minute chunks. For the test data, we assume that we only know some of the 15-minute chunks. Our target is to predict the occupancy in the rest of the day, or 30 minutes ahead.

5.1 Occupancy Prediction of Individuals

The individual occupancy prediction results on datasets Study10 and Study11 are summarized in Table 1. In Study10, the mixture EGH performs better than kNN for the occupancy prediction for person2 on 02/17/2014. Furthermore, mixture EGH outperforms kNN for both persons on 02/20/2014. However, for person1 on 02/19/2014, kNN works a little bit better. When checking the original data on this test date, we find that the activities on this date are very similar

Table 1: Precision Recall F-measure Comparison of Individual and Whole House Occupancy Prediction in Study 14.

Dataset	Date	Person	EGH			kNN			SVM		
			precision	recall	fmeasure	precision	recall	fmeasure	precision	recall	fmeasure
study10	02/17/2014	person2	**1.00**	**1.00**	**1.00**	0.99	0.98	0.98	0.71	0.76	0.71
	02/19/2014	person1	0.98	0.99	0.98	**0.99**	**0.99**	**0.99**	0.71	0.76	0.70
	02/20/2014	person2	**0.93**	**0.92**	**0.92**	0.92	0.91	0.90	0.72	0.77	0.72
	02/20/2014	person1	**0.95**	**0.94**	**0.94**	0.94	0.93	0.93	0.71	0.77	0.72
	02/20/2014	**wholehouse**	**0.92**	**0.92**	**0.91**	0.91	0.89	0.91	0.79	0.74	0.74
study11	02/04/2014	person2	0.93	0.93	0.92	**0.95**	**0.95**	**0.95**	0.71	0.77	0.72
	02/04/2014	person1	0.93	0.93	0.92	**0.95**	**0.95**	**0.95**	0.70	0.77	0.71
	02/05/2014	person2	**0.85**	**0.92**	**0.86**	0.87	0.87	0.84	0.71	0.76	0.71
	02/05/2014	person1	**0.84**	**0.90**	**0.84**	0.79	0.90	0.80	0.70	0.77	0.71
	02/04/2014	**wholehouse**	**0.918**	**0.924**	**0.913**	0.916	0.921	0.911	0.77	0.69	0.71
	02/05/2014	**wholehouse**	**0.90**	**0.84**	**0.84**	0.88	0.81	0.81	0.74	0.70	0.70

to the historical activities in the training data. This observation leads us to the conclusion that when the test data is very highly similar to the historical data, the kNN approach sometimes performs a little better. In Study11, mixture EGH gets higher precision, recall and f-measure scores on 02/05/2014, but the opposite is true on 02/04/2014. We analyzed the original data to find the reason for the kNN's better performance, and found the date is an anomaly from the normal pattern, since both individuals went to sleep late that day (after 12:00am). Before sleep, person1 even stayed in the kitchen for around two hours. The frequent episode KZ, which represents 'kitchen-unoccupied', usually occurs in the morning instead of around midnight. However, the mixture EGH model still assumes that the KZ pattern happens during the morning; therefore the prediction results are not accurate. Since kNN ignores this fine granular activity pattern at a house and only considers the occupancy status in the past most similar five days, its performance is better. Generally speaking, the mixture EGH helps predict when a person leaves home and the period of sleeping and its performance is competitive to the kNN approach. For all these experiments, the SVR approach performs the worst because of the limitations of this approach. SVR uses the latest several data points about the occupancy state as the training vector for the prediction of the next occupancy state. Here we set eight past data points as the predictor. Other features such as the time of day and day of week cannot be utilized fully.

We also conduct experiments for individuals' *rest-of-day* occupancy prediction at different times. Figure 4 illustrates a person's occupancy prediction result in Study10. There are three sub-figures. Each sub-figure describes the precision, recall, and f-measure of $person1$ on 02/20/2014. The blue line represents the mixture EGH model, the green line represents the PDF model, and the red line denotes the kNN model. The x-axis is the number of known 15-minute chunks of the test day. For instance, at $x = 20$, we already know $20 * 15$ minutes' data and need to predict whether the home is occupied during the remaining 76 chunks. The y-axis denotes the precision, recall and f-measure values in the three sub-figures from the top down. The first sub-figure shows that the mixEGH has the highest precision, recall and f-measure on test day 02/20/2014 for occupancy prediction. The other two baseline approaches are comparable, except that kNN performs better than PDF when the person comes back home after slot 72. Looking into the original data, we find that person1 actually comes home later than usual in the training dataset.

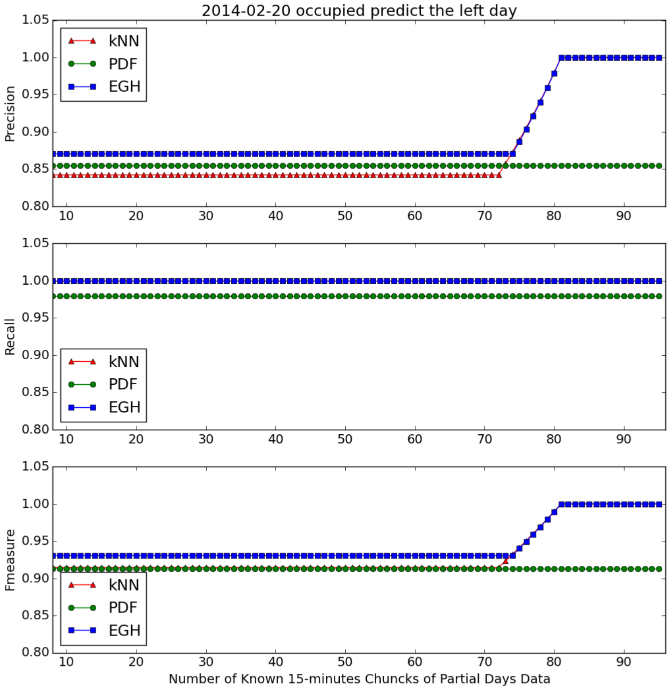


Figure 4: Occupancy prediction precision, recall and f-measure comparison of three approaches of person1 on 02/20/2014 on Study10.

5.2 Occupancy Prediction of Residential Buildings

Based on individual prediction results, we deduce when a house is occupied using logic OR operations on the prediction results of two persons. The whole house occupancy prediction results are listed in Table 1 and marked in bold. In Study10, the precision, recall, and fmeasure values of the whole house are 0.92, 0.92 and 0.91, respectively, which are higher than the values from the kNN approach of 0.91, 0.90 and 0.91, and of the SVR approach 0.79, 0.74 and 0.74. Similarly, the mixture EGH model outperforms kNN in Study11. Note that in Study 11 on 02/04/2015, EGH does not perform as well as kNN on individuals but performs a little bit better than kNN, and much better than SVR in occupancy prediction of the whole house. The reason behind this is because the activities of the two people inside the home are not synchronized. The mixture EGH model can predict the occupancy for each person and grasp each person's activities more accurately.

5.3 Limitations of Mixture EGH Model

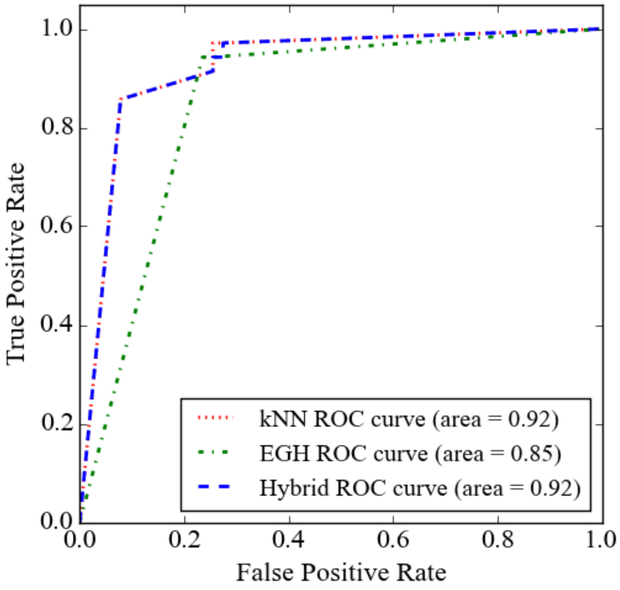
Although the temporal mixture EGH model performs well on the datasets Study10 and Study11, the same is not true for the dataset Study14. Table 2 shows that, in Study14, the mixture EGH model works better for the individual and whole-house occupancy predictions on 12/18/2013 but not on 12/19/2013 or 12/20/2013. We check the activities of both individuals on these two days and find that both of them went out again after coming back and staying home for a while. Since the episodes of going out after coming back home from work do not occur frequently, mixture EGH cannot detect this pattern. Thus the occupancy prediction probability of these events is completely missing. However kNN performs well because it leverages all the historical data; therefore, even if the abnormal event occurs once, this prediction approach incorporates it and obtains the average value. To relax the limitations of abnormal events, we propose a hybrid model for prediction; when deploying this occupancy prediction in reality, for example for a prediction that is 30 minutes ahead, just 15 minutes before the prediction, if a person goes out again after coming back, the deployed system switches to the kNN approach rather than the mixture EGH model for prediction. In such cases, this hybrid model can always get the best prediction results.

5.4 House Occupancy Prediction 30 Minutes Ahead with Hybrid Approach

To preheat the house, we need to evaluate how much time in advance to automatically turn on/off HVAC, and the advance notice time estimation is given in [13]. Here we use prediction of 30 minutes ahead of time house occupancy. We compare the receiver operating characteristic (ROC curve) of three approaches: mixture EGH model, kNN, and a hybrid approach of the mixture EGH model and kNN. In this hybrid approach, we set mixture EGH results as the baseline, then replace the values of the mixture model by the values from kNN model in the following two situations: 1) After a person comes back home; and 2) When the prediction probability of kNN is greater than 0.8. Figure 5, 6, and 7 illustrate the ROC curve of the whole house occupancy prediction on 02/20/2014 of dataset Study10, 02/04/2014 of dataset Study 11 and 12/20/2013 of dataset Study14, respectively. The red and green lines represents the kNN and mixture EGH models; the blue line denotes the hybrid approach. The ROC curves show that the hybrid approach always has the largest area, namely 0.96, 0.92 and 0.92, which indicate that the hybrid approach always performs best.

6 CONCLUSION

Residential occupancy prediction is a hot research topic in controlling HVAC systems. The accuracy of occupancy prediction influences the comfort of the persons inside the home and energy savings. In order to achieve the highest prediction result, we propose to integrate the mixture EGH model and kNN together as a hybrid approach.



(a)

Figure 5: ROC curve of house occupancy prediction in Study10 (02/20/2014).

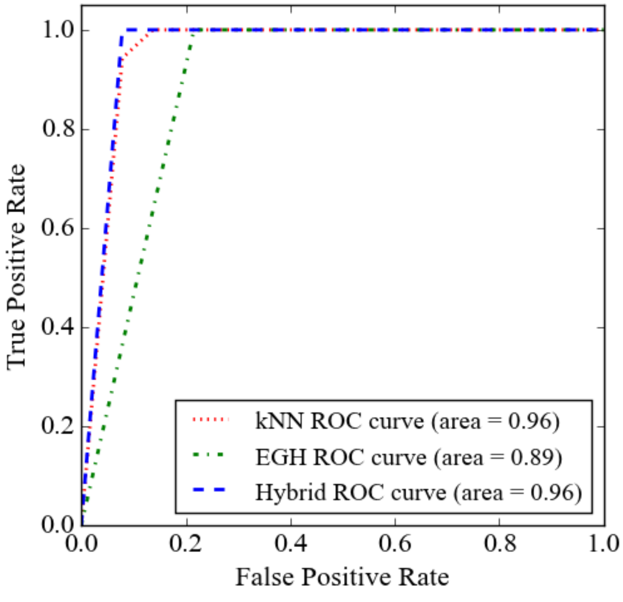
Our work differs from previous research based on the main contributions listed below:

(1) We formulate the problem as one of temporal mining; the activities inside the building are abstracted as episodes, and each episode is connected with an episode generative HMM model.
(2) We mine the activity patterns according to the time and gap: both the duration of each type of activity, and the gap between two consecutive events are limited in a proper range. This range is extracted from the historical data according to the weekday and holidays.
(3) Our hydrate prediction solution performs best on the workday occupancy prediction: in case of normal activities, we apply mixture EGH model; in case of abnormal events, we utilize kNN, which is generally considered a benchmark in occupancy prediction problem.

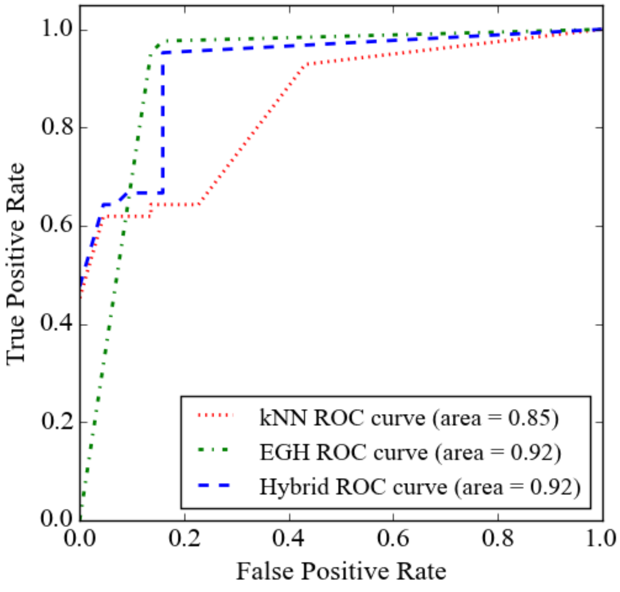
In the future work, we will continue working on the holiday occupancy prediction. The occupancy patterns for these days are completely different. For example, on certain weekdays, a person may never go out. Therefore the occupancy prediction probably depends more on date than the indoor activities. Furthermore, we will apply this temporal mining approach to the GPS datasets [6] to check the effectiveness of occupancy prediction with different kinds of data.

Table 2: Precision Recall F-measure of Individual and Whole House Occupancy Prediction in Study 14.

Dataset	Date	Person	EGH			kNN			SVM		
			precision	recall	fmeasure	precision	recall	fmeasure	precision	recall	fmeasure
study14	12/18/2013	person2	**0.91**	**0.91**	**0.89**	0.87	0.87	0.84	0.73	0.77	0.71
	12/18/2013	person1	**0.92**	**0.92**	**0.91**	0.90	0.90	0.89	0.73	0.76	0.71
	12/19/2014	person2	0.86	0.86	0.85	**0.90**	**0.90**	**0.88**	0.73	0.76	0.71
	12/19/2014	person1	0.85	0.84	0.84	**0.86**	**0.86**	**0.85**	0.73	0.76	0.71
	12/20/2014	person2	0.92	0.94	0.92	**0.98**	**0.97**	**0.97**	0.75	0.79	0.75
	12/20/2014	person1	0.90	0.91	0.90	**0.95**	**0.95**	**0.95**	0.75	0.79	0.75
	12/18/2013	**wholehouse**	**0.91**	**0.91**	**0.90**	0.88	0.88	0.86	0.75	0.72	0.70
	12/19/2013	**wholehouse**	0.841	0.845	0.838	**0.848**	**0.853**	**0.842**	0.79	0.74	0.74
	12/20/2013	**wholehouse**	0.92	0.90	0.90	**0.94**	**0.93**	**0.93**	0.74	0.72	0.70



((b)



(c)

Figure 6: ROC curve of house occupancy prediction in Study11 (02/04/2014).

Figure 7: ROC curve of house occupancy prediction in (c) Study14 (12/20/2013).

REFERENCES

[1] Abdullah Alrazgan, Ajay Nagarajan, Alexander Brodsky, and Nathan E Egge. 2011. Learning occupancy prediction models with decision-guidance query language. In *Proceedings of the 44th Hawaii International Conference on System Sciences (HICSS)*.

[2] Alex Beltran and Alberto E Cerpa. 2014. Optimal HVAC building control with occupancy prediction. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*.

[3] Varick L Erickson and Alberto E Cerpa. 2010. Occupancy based demand response HVAC control strategy. In *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*.

[4] Niko Kiukkonen, Jan Blom, Olivier Dousse, Daniel Gatica-Perez, and Juha Laurila. 2010. Towards rich mobile phone datasets: Lausanne data collection campaign. *Proc. ICPS, Berlin* (2010).

[5] Wilhelm Kleiminger, Friedemann Mattern, and Silvia Santini. 2014. Predicting household occupancy for smart heating control: A comparative performance analysis of state-of-the-art approaches. *Energy and Buildings* (2014).

[6] Christian Koehler, Brian D Ziebart, Jennifer Mankoff, and Anind K Dey. 2013. TherML: occupancy prediction for thermostat control. In *Proceedings of the 2013 ACM international Joint Conference on Pervasive and Ubiquitous Computing*.

[7] John Krumm and AJ Bernheim Brush. 2011. Learning time-based presence probabilities. In *Pervasive Computing*.

[8] Srivatsan Laxman, Vikram Tankasali, and Ryen W White. 2008. Stream prediction using a generative model based on frequent episodes in event sequences. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[9] Sawsan Mahmoud, Ahmad Lotfi, and Caroline Langensiepen. 2013. Behavioural pattern identification and prediction in intelligent environments. *Applied Soft Computing* (2013).

[10] Carlo Manna, Damien Fay, Kenneth N Brown, and Nic Wilson. 2013. Learning occupancy in single person offices with mixtures of multi-lag Markov chains. In *Proceedings of the IEEE 25th International Conference on Tools with Artificial Intelligence (ICTAI)*.

[11] Heikki Mannila, Hannu Toivonen, and A Inkeri Verkamo. 1997. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery* (1997).

[12] Debprakash Patnaik, PS Sastry, and K Unnikrishnan. 2008. Inferring neuronal network connectivity from spike data: A temporal data mining approach. *Scientific Programming* (2008).

[13] James Scott, AJ Bernheim Brush, John Krumm, Brian Meyers, Michael Hazas, Stephen Hodges, and Nicolas Villar. 2011. PreHeat: controlling home heating using occupancy prediction. In *Proceedings of the 13th International Conference on Ubiquitous Computing*.

[14] Huijuan Shao, Manish Marwah, and Naren Ramakrishnan. 2013. A Temporal Motif Mining Approach to Unsupervised Energy Disaggregation: Applications to Residential and Commercial Buildings. Bellevue, U.S.A.