Towards Interactive Learning for Occupancy Estimation

Manar Amayri, Stéphane Ploix, Patrick Reignier, Sanghamitra Bandyopadhyay

TOWARDS INTERACTIVE LEARNING FOR OCCUPANCY ESTIMATION

Manar Amayri[1], Stéphane Ploix[1], Patrick Reignier[2], Sanghamitra Bandyopadhyay[3]
[1]G-SCOP lab, Grenoble Institute of Technology, Grenoble, France
[2]LIG lab, Grenoble Institute of Technology, Alpes, Grenoble, France
[3]Indian Statistical Institute, Kolkata, India

Abstract

A new kind of supervised learning approach is proposed to estimate the number of occupants in a room. It introduces the concept of interactive learning where actual occupancy is interactively requested to occupants when it is the most valuable to limit the number of interactions. Occupancy estimation algorithms rely on machine learning: they use information collected from occupants together with common sensors measuring motion detection, power consumption or CO_2 concentration for instance. Two different classifiers are considered for occupancy estimation with interactions: a decision tree C4.5 classifier and parameterized rule based classifier. In this paper, the question of when asking to occupants is investigated. This approach avoids the usage of a camera the determine the actual occupancy.

Keywords: interactive learning, human behavior, building performance, office buildings, machine leaning and data mining

Type of Submission/Paper: Regular Research Papers

1 Introduction

Recently, research about building turns to focus on occupant behavior. Interactive learning opens the gate of the involvement of occupants through the exchange of information. It is a new challenge in occupancy estimation because it solves the issue of the measurement of actual occupancy usually done thanks to cameras and a posteriori labeling, which is both time consuming and invasive. Interactive learning has not been investigated up to now and should be considered as a new approach, that has been applied for occupancy estimation. Most of the works deal with the design stage: the target is to represent the diversity of occupant behavior in order to improve building energy manegment capabilities . Most of the approaches use statistics about human behavior (Roulet et al., 1991; Page et al., 2007; Robinson and Haldi, 2009). (Kashif et al., 2013) emphasized that inhabitants' detailed reactive and deliberative behavior must also be taken into account and proposed a co-simulation methodology to find out the impact of certain actions on energy consumption. Nevertheless, human behavior is not only interesting during the design step, but also during operation. It is indeed useful for diagnostic analyses to discriminate human misbehavior from building system performance, and also for energy management where strategies depend on human activities and, in particular, on the number of occupants in a zone. Such a system as to be trained in each new environment. Unfortunately, using supervised learning algorithm on site is not widely accepted because of the required target to build the set of training data , which usually come from cameras which are not acceptable for many users. In addition, labeling occupancy from videos is time consuming.This paper tackles this issue. It proposes an occupancy estimation approach based on interactive learning with occupants in the studied area. Section 2 presents a state of the art about occupancy estimation. Section 3 discusses the proposed process of interactive learning that interacts with occupants to collect the actual occupancy. Section 4 focuses on C45 decision tree classifier for interactive learning. Section 5 focuses on parameterized classifier for interactive learning and section 6 compares the two classifiers in interactive learning context in an office.

2 State of the art

Different approaches for estimating occupancy have been investigated but still without using an interactive learning process. Methods vary from basic single feature classifiers that distinguish among two classes (presence and absence) to multi-sensor, multi-feature models. A primary approach, which is prevalent in many commercial buildings, is the usage of passive infrared (PIR) sensors for occupancy. However, motion detectors fail to detect a presence when occupants remain relatively still, which is quite common during activities like working on a computer or regular desk work. This makes the use of only PIR sensors for occupancy counting purpose less attractive. Conjunction of PIR sensors with other sensors can be useful as discussed in (Agarwal et al., 2010). It uses motion sensors and magnetic reed switches for occupancy detection to increase the efficiency of HVAC systems in smart buildings, which is quite simple and non-intrusive. Apart from motion, acoustic sensors (Padmanabh et al., 2009) may be used. However, audio from the environment can easily fool such sensors, and with no support from other sensors, it can report many false positives. In the same way, other sensors like video cameras (Milenkovic and Amft, 2013b), which exploit the huge advances in the field of computer vision and the ever increasing computational capabilities, RFID tags (Philipose et al., 2004) installed on id cards, sonar sensors (Milenkovic and Amft, 2013a) plugged on monitors to identify presence of a person on a computer, have been used and have proved to be much better at solving the problem of occupancy count, yet can not be employed in most office buildings for reasons like privacy and cost concerns. The use of pressure and PIR sensors to determine presence/absence in single desk offices has been discussed in (Nguyen and Aiello, 2012); it further tags activities based on this knowledge. However, for various applications like activity recognition or context analysis within a larger office space, information regarding the presence or absence of people is not sufficient and an estimation of the number of people occupying the space is essential. (Lam et al., 2009) investigates this problem in open offices, estimating occupancy and human activities using a multitude of ambient information, and compare the performance of HMMs, SVMs and Artificial Neural Networks. However, none of these methods generate human-understandable rules which may be very helpful to building managers. In general, an occupancy count algorithm that fully exploits information available from low cost, non-intrusive, environmental sensors and provides meaningful information is an important yet little explored problem in office buildings. This occupancy detection systems still have certain limitations with respect to occupant privacy.

3 Principle of interactive learning

A new methodology for occupancy estimation has been investigated by using interactive learning approach.

Interactive learning is a process involving exchange of information with the users in order to collect some important data according to the problem context. In supervised learning methods, which are used widely in a lot of applications, the problem of the required target arises in the estimation of the number of occupants i.e. the labeling issue usually have been taken from installed video cameras. Using camera is still not acceptable in many places for respecting the privacy of occupants. Interactive learning is an extension of supervised learning machine that in our case will estimate the occupancy by collecting the required labeling from the occupants themselves. The problem statement of occupancy estimation could be formalized like this: let $(A_{1,k}, A_{2,k}, \ldots, A_{n,k}, C_k)$ be a *record* where $A_{i,k}$ is an attribute value, i.e. feature or sensor data, for record k, and C_k a belonging class, i.e. the actual number of occupants provided thanks to interactions here. $(A_{1,k}, A_{2,k}, \dots, A_{n,k})$ is named an *ask*: it is an incomplete record. A classifier is defined over $D = dom(A_{1,k}) \times dom(A_{2,k}) \times \cdots \times dom(A_{n,k})$ with $dom(A_{i,k}) = [\check{A_i}, \hat{A_i}]$ where $\check{A_i}$ and $\hat{A_i}$ stand respectively for the maximum and the minimum value of the recorded attributes. It leads to: $\forall i, (A_{1,k}, A_{2,k}, \dots, A_{n,k}) \in D_i(\Theta) \to C_k = class_i$ with $\{D_i(\Theta); \forall i\}$, a partition of the attribute space D corresponding of $class_i$. Θ is the list of the classifier parameters. Asking problem is the main issue in interactive learning to define when it is necessary to ask and when asking is useless. Here, the interaction will be with people to give the actual number of occupants, To perform the task of asking the number of occupants, let $\{(A_{1,k}, A_{2,k}, \dots, A_{n,k}, C_k); \forall k\}$ be a set of records. Let $\{D_i(\Theta); \forall i\}$ be a parameterized classifier with Θ the current parameter values of the classifier. Wherever the model of the classifier is more complicated the parameterized classifier will be more difficult to analyze.

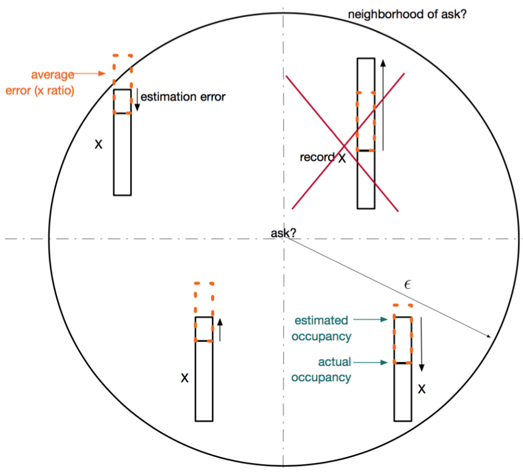

Figure 1: Asking problem

The asking problem consists in determining a utility function for an ask $(A_{1,k'}, A_{2,k'}, \dots, A_{m,k'})$ taking into account the already available records and the classifier. Three criteria are taken into account to determine whether an ask is potentially useful or not:

1. the density of the neighborhood. It is the number of existing records in the neighborhood of the potential ask. The neighborhood is defined by the distance $\left\|\left(\frac{A_{1,k'}-A_{1,k}}{\hat{A_1}-\check{A_1}}, \frac{A_{2,k'}-A_{2,k}}{\hat{A_2}-\check{A_2}}, \dots\right)\right\|_\infty < \frac{\epsilon}{2}$ where $\hat{A_i}$ and $\check{A_i}$ stand respectively for the maximum and the minimum value of the recorded attributes A_i. The neighborhood can be modified according to the $\epsilon \in [0, 1]$. The local record density in the neighborhood of an ask $(A_{1,k'}, A_{2,k'}, \dots)$ is given by equation (1); $(A_{1,k}, A_{2,k}, \dots, C_k)$ stands for existing record k.
2. the classifier estimation error in the neighborhood of the potential ask that leads to the concept of quality neighborhood. As shown in figure 1, if the classifier estimation error is too high for a record, it is removed from the neighborhood because of the poor quality. A record $(A_{1,k}, A_{2,k}, \dots, C_k)$ is considered in the quality neighborhood of a potential ask if error is small i.e. $|average(C_k) - occupancy_k| < E_r\zeta$ where $average(C_k)$ stands for the average occupancy of the class C_k, $occupancy_k$ is the actual recorded occupancy collected thanks to an ask, $E_r \in [1, 2)$ typically, is an error ratio than can be adjusted, ζ is the average estimation error of the n existing records: $\frac{1}{n}\sum_{k=0}^{n-1} |average(C_k) - occupancy_k|$.
3. the minimum class weight i.e. the minimum number of records for each class: $weight(class_x) = \left|\{(A_{1,k}, A_{2,k}, \dots, C_k); C_k = class_x\}\right|$. In case of an potential *ask* $(A_{1,k'}, A_{2,k'}, \dots)$, the ask will contribute to a class determined by the classifier because ask has not been performed up to now:

$$class = D_{k'}(\Theta, (A_{1,k'}, A_{2,k'}, \dots))$$

The minimum class weight $weight(class_x) < C_w$ can be adjusted according to the problem.

All the potential asks that satisfy the above three criteria are asked to the occupants in order to possibly become an additional record.

if record density in quality neighborhood of a potential ask (neighborhood without

records with big estimation error) is low or $weight(class_x) \leq C_w$ **then**
  Ask
else
  Do not Ask
end if

$$d(A_{1,k'}, A_{2,k'}, \dots) = \frac{\left| \left\{ \begin{array}{c} (A_{1,k}, A_{2,k}, \dots, C_k); \dots \\ (A_{1,k}, A_{2,k}, \dots) \in \dots \\ \left\| \left(\frac{A_{1,k'} - A_{1,k}}{\hat{A_1} - \check{A_1}}, \frac{A_{2,k'} - A_{2,k}}{\hat{A_2} - \check{A_2}}, \dots \right) \right\|_\infty < \frac{\epsilon}{2}, \\ \dots \epsilon \in (0,1], \forall k \end{array} \right\} \right|}{n} \tag{1}$$

For validation, occupant reaction has to be taken into account as a response probability p i.e. whether the occupants answer or not. For a given context, the number of asks depends on the classifier used for occupancy estimation:

- C45 decision tree classifier can be used directly together with the ask mechanism to generate occupancy labels for preparing training data .
- a parameterized classifier can be used as well where parameters can be adjusted according to the growing number of records.

These classifiers together with the proposed ask mechanism are presented in the 2 next sections.

4 Decision tree classifier

The test bed is an office in Grenoble Institute of Technology, which accommodates a professor and 3 PhD students. The office has frequent visitors with a lot of meetings and presentations all through the week. The set-up for the sensor network includes:

- 2 video cameras for recording real occupancy numbers and activities. Those two cameras are only used for validation purpose.
- An ambiance sensing network, which measures luminance, temperature, relative humidity (RH), motions, CO2 concentration, power consumption, door and window position, microphone. Data are sent thanks to ENOCEAN protocol on significant value change event.
- A centralized database with a web-application for retrieving data from different sources continuously.

To perform the task of finding the number of occupants, a relation has to be discovered between the office environment and the number of people in it . The office environment can be represented as a set of state variables, $A_t = [A_1, A_2, \dots, A_m]_t$. This set of state variables A at any instance of time t must be indicative of occupancy. A state variable can be termed as a feature, and therefore the set of features as feature vector. Similarly, the m-dimensional space that contains all possible values of such a feature vector is the feature space. The underlying approach for the experiments is to formulate the classification problem as a map from a feature vector into some feature space that comprises several classes of occupancy or activities. Therefore, the success of such an approach heavily depends on how good the selected features are. In this case, features are attributes from multiple sensors accumulated over a time interval. The choice of interval duration is highly context dependent, and has to be done according to the granularity required.

Features is the information extracted from the data i.e acoustic pressure from a microphone, time slot, occupancy from power consumption, door or window position, motion counting,day type, indoor temperature,... One quantitative measurement of the usefulness of a feature is *information gain*, which depends on the concept of *entropy* (Amayri et al., 2015). Information gain is helpful to distinguish among a large set of features, the most worthwhile to consider for occupancy estimation.

A supervised learning approach has been used. Occupancy has been determined before using a classification algorithm: occupancy counting was manually annotated using a video feed from two cameras strategically positioned in an office to simulate the occupant replies, determine the structure of parameterized classifier and validate interactive learning results.

The *decision tree* classification technique has been selected because it provides both very good results and the results are easy to analyze and adapt. The decision tree algorithm selects a class by descending a tree of decision nodes. Each internal node represents a comparison of a single feature value with a learned threshold. The target of the decision tree algorithm is to select features that are more useful for classification. Finally, because decision trees are human readable, they can be adjusted using expert knowledge and extract the estimation rules (if-then) from the decision tree structure.

if $X_i \leq \ threshold$ **then**
  left child node
else
  right child node
end if

5 Parameterized classifier

Another approach for occupancy estimation is investigated because it fits well with interactive learning. It uses a predetermined classifier structure with parameters to be adjusted according to the incoming records. Any classifier could be used in this approach, but still it is important to choose a general structure for the sake of adaptability. Additionally, the number of parameters should be low because the tuning mechanism relies on an optimization process that may become inefficient when complexity increases. A depth-limited *decision tree* classifier has been selected here. The depth of a tree varies depending upon the size and nature of the sample set. For example if the depth of the tree is set to '1', a tree with a single node is generated. Otherwise, the most complicated case builds a complete tree, where every path test every feature. Limiting the depth avoids data over-fitting phenomena by rejecting non significant features. Assuming n_s samples and n_f features, at each level (i), the remaining $(n_f - i)$ features for each sample at the level (i) should be examined to calculate the information gain. However, learned trees are rarely complete (number of leaves is lower or equal to n_s). In practice, complexity is linear in both number of features (n_f) and number of training samples (n_s). In addition, a maximum tree depth of d will limit the maximum number of rules for a decision to d. In general, a deep tree with many leaves is usually highly accurate on the training data but less with the validation data. In addition, finding a shortest decision tree is preferred over longer trees: it is indeed easier to understand and more reliable, it is also easier to implement and to use. Tuning problem can be solved by adjusting the classifier parameters (node thresholds of the decision tree) in the final structure according to each updated record set and how much it's different from the previous one. An objective function is determined to minimize the distance between actual (coming from an ask) and estimated (coming from the classifier) number of occupants in the room. Optimization covers a required period of asking, interacting with the occupants in the studied area.

A depth equal to 2 is the limitation chosen for the next analysis of occupancy estimation because of the low average error of the resulting decision tree and of the little number of thresholds to adjust. Additionally, the tree is readable and rules are quite general as it is shown in figure 2.

if microphone is low **then**
  ≈ 0 person
else if microphone is high and CO2 physical model is low **then**
  ≈ 1 person
else if microphone is high and CO2 physical model is high **then**
  ≈ 2 persons
end if

Note that, (if-then) rules from the tree structure could be extracted now easily to be applied in a tuning context.

6 Results

The data covers 11 days from 04-May-2015 to 14-May-2015. During these 10 days, an Human Machine Interface (HMI) is assumed to be used to interact with the users in the office.

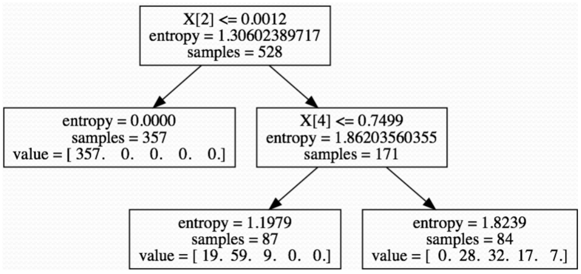

Figure 2: Decision tree used by the parameterized classifier

In this HMI, an alarm is triggerer to ask the user the actual number of occupants. The replies of the occupants are modeled but a random process with a reply probability 50% i.e. only half of the asks get replies.

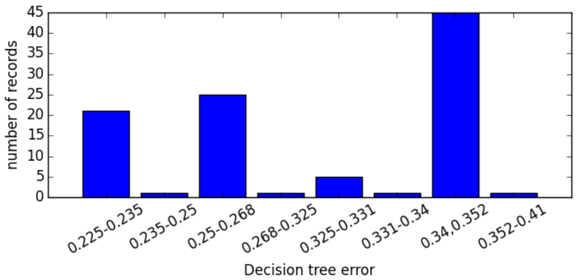

Figure 3: Distribution of decision tree error with 13 asking

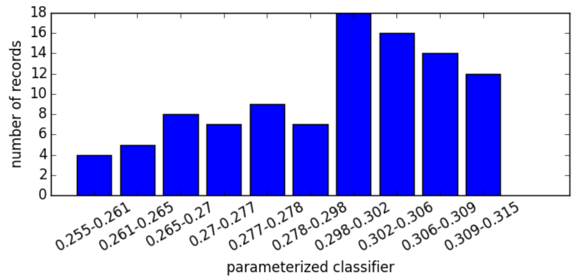

Figure 4: Distribution of parameterized classifier error with 13 asking

Both estimation methods have been applied. The interactive learning process has been performed 100 times to show the distribution of the error for both *decision tree* (see figure 3) and parameterized classifier (see figure 4), with 13 asks, $\epsilon = 0.5$, and $E_r = 1.5$. For the first 13 asks, the parameterized classifier is giving better results than the whole decision tree. While reproducing the same interactive learning process with 50 asks, figures 5 and 6 with $\epsilon = 0.5$, and $E_r = 1.5$ are obtained. The decision tree starts to give better results. However *decision tree* needs more than 13 asks for training data to build an acceptable estimator. Additionally, it is important to notice that increasing ϵ decreases the number of asks while increasing the error ratio decreases the number of asks. It can be noticed also that the asking process is dependent on the classifier used because the estimation error intervenes. The following table illustrates how the 13 asks are distributed along the days with parameterized estimator. Asking process with decision tree leads to almost the same results depending of the run that contains randomness because of the ask replies.

Day	1	2	3	4	5	6	7	8	9	10
Asking	9	3	0	0	0	0	0	1	0	0

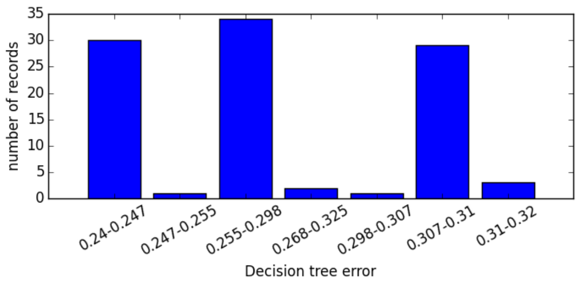

Figure 5: Distribution of decision tree error with 50 asking

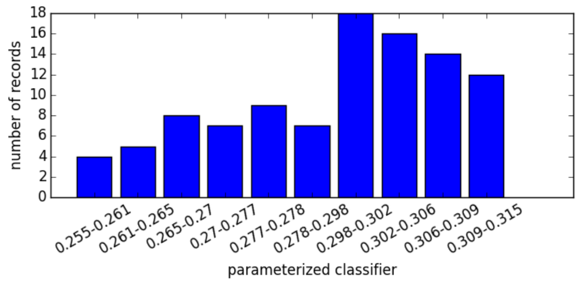

Figure 6: Distribution of parameterized classifier error with 50 asking

7 Conclusion

An interactive learning approach has been proposed in this paper to avoid the manual labeling of actual occupancy in a room for supervised learning approaches. It is then possible with a little number of interactions with occupants to estimate the number of occupants in a zone. Two different classifiers have been tested together with the interactive ask process: a pure C45 decision tree algorithm and parameterized rule based classifier. The approach can be easily extended to any kind of classifier.

The C45 decision tree algorithm is very general because its structure is not assumed: it is discovered from the data and can be therefore extended to any room with any sensors. It leads to the best results after about 13 asks. The parameterized classifier yields better results at first but because the number of parameters (2) is much less than the decision tree (about 45 parameters), the decision tree finally better estimates the number of occupants although the parameterized classifier directly minimizes the estimation error and the classification (with C45, classifying in class 2 or in class 3 instead of class 1 has the same impact). Because the structure of parameterized classifier is predefined, the adaptation capability to another context is much less: a relevant structure has to be proposed.

The impact of the modality of interactions through the human machine interface has still to be investigated.

8 Acknowledgment

The authors acknowledge the support of the French Agence Nationale de la Recherche (ANR) under reference ANR-13-VBDU-0006 (OMEGA project).

Acknowledgment

The authors acknowledge the support of the French Agence Nationale de la Recherche (ANR) under reference ANR-13-VBDU-0006 (OMEGA project).

REFERENCES

Agarwal, Y., Balaji, B., Gupta, R., Lyles, J., Wei, M., Weng, T., 2010. Occupancy-driven energy management for smart building automation. In: Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building. ACM, pp. 1–6.

Amayri, M., Ploix, S., Bandyopadhyay, S., 2015. Estimating occupancy in an office setting. ASCE conference 2015, Pittsburgh, USA.

Kashif, A., Dugdale, J., Ploix, S., 2013. Simulating occupants' behaviour for energy waste reduction in dwellings: A multi agent methodology. Advances in Complex Systems 16, 37.

Lam, K. P., Höynck, M., Dong, B., Andrews, B., shang Chiou, Y., Benitez, D., Choi, J., 2009. Occupancy detection through an extensive environmental sensor network in an open-plan office building. In: Proc. of Building Simulation 09, an IBPSA Conference.

Milenkovic, M., Amft, O., 2013a. An opportunistic activity-sensing approach to save energy in office buildings. In: Proceedings of the fourth international conference on Future energy systems. ACM, pp. 247–258.

Milenkovic, M., Amft, O., 2013b. Recognizing energy-related activities using sensors commonly installed in office buildings. Procedia Computer Science 19, 669–677.

Nguyen, T. A., Aiello, M., 2012. Beyond indoor presence monitoring with simple sensors. In: PECCS. pp. 5–14.

Padmanabh, K., Malikarjuna V, A., Sen, S., Katru, S. P., Kumar, A., Vuppala, S. K., Paul, S., et al., 2009. isense: a wireless sensor network based conference room management system. In: Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings. ACM, pp. 37–42.

Page, J., Robinson, D., Scartezzini, J., 2007. Stochastic simulation of occupant presence and behaviour in buildings. Proc. Tenth Int. IBPSA Conf : Building Simulation, 757–764.

Philipose, M., Fishkin, K. P., Perkowitz, M., Patterson, D. J., Fox, D., Kautz, H., Hahnel, D., 2004. Inferring activities from interactions with objects. Pervasive Computing, IEEE 3 (4), 50–57.

Robinson, D., Haldi, F., 2009. Interactions with window openings by office occupants. Energy and Buildings 44, 2378–2395.

Roulet, C., Fritsch, R., Scartezzini, J., Cretton, P., 1991. Stochastic model of inhabitant behavior with regard to ventilation. Technical report.