Work in Progress session at PerCom 2012, Lugano (19-23 March 2012)

Occupancy Detection in Commercial Buildings using Opportunistic Context Sources

Sunil Kumar Ghai†, Lakshmi V Thanayankizil‡, Deva P. Seetharam†, Dipanjan Chakraborty†
†*IBM Research, India,* ‡*Georgia Institute of Technology, USA*
{*sunilkrghai, dseetharam, cdipanjan*}*@in.ibm.com, lakshmi@gatech.edu*

***Abstract*—Accurate occupancy information in commercial buildings can enable several useful applications such as energy management and dynamic seat allocation. Most prior efforts in this space depend on deploying an additional network of deeply coupled sensors to gather occupancy details. This paper presents a novel approach for occupancy detection using only context sources that are commonly available in commercial buildings such as area access badges, Wi-Fi access points, Calendar and Instant Messaging clients. We present models to conduct a situation-centric profiling using such sources and evaluate results of those models. Through a pilot study of a building floor with 5 volunteers for 6 weeks, we demonstrate the potential for detecting occupancies with accuracy as high as 90%.**

***Keywords*-Context Aware, Energy Management, Smart Buildings, Occupancy Detection, Soft Sensors**

I. Introduction

Accurate occupancy information in commercial buildings can enable several useful applications such as energy management and dynamic seat allocation. Researchers have proposed a number of schemes [1], [2] to perform occupancy-driven control of electrical loads using various sensors (for example, cameras, augmented PIRs, CO_2 sensors, etc). Although recent advances in wireless sensor networking technologies reduce the wiring cost, costs for deployment and management of a parallel sensor infrastructure remain significant and limit the widespread adoption of these schemes.

We approach the problem differently. We observe that many office buildings today are equipped with computing (Wi-Fi access points and mobile computing devices) and security (access control systems) infrastructure that can provide valuable cues for detecting occupancy. We call such context sources as *opportunistic context sources or soft-sensors*. Occupancy detection using such soft-sensors can be economical and easier to adopt as it does not require installation of additional infrastructure. It is important to note that this approach does not preclude additional occupancy sensors but can incorporate data from hardware sensors in locations where the software sensors prove to be inadequate.

The key contributions of this paper are two fold. First, applying well-known machine learning techniques (regression and classification) to develop a cost-effective system for inferring occupancy using opportunistic context sources; second, evaluating the feasibility of the proposed approach through a pilot trial conducted in an office for a period of six weeks. The rest of the paper is organized as follows. Section II gives an overview of the pilot and Section III identifies system challenges. Section IV presents the details of these two approaches and in Section V we present inference results. Then we review existing body of work in Section VI before we conclude in Section VII.

II. Pilot Overview

The pilot trial was conducted in a commercial building by tracking five volunteers working on an office floor. The floor consists of multiple open workstations and eleven meeting rooms. A monitoring agent was installed on the volunteers' laptops[1] and to collect the data once every minute from the following: *Wi-Fi Access Points, User Calendar, System Activity Monitor, Instant Messaging (IM) Clients and Time-of-day.* Further details on each context source and the type of context cues they provide are summarized in Table I.

Ground Truth Establishment: For system validation (and for training), the volunteers manually tagged (using a desktop application installed on their laptops) their current location whenever they moved. For example, when they were working in their cubicle, they entered the tag *cubicle*, while attending a meeting, the tag *meeting with the specific room number* and the tag *break*, when they took a break. These tags were considered as the 'Ground Truth' while evaluating the system.

III. Challenges

Many of the aforementioned context sources (for example, IM, Calendar) are meant for other purposes (e.g. chatting or maintaining appointments) and are at best fuzzy indicators of an employee's state. Some of the challenges in working with such sources are:

Unreliability: Context sources, like IM status, Calendar rely on the employees to get updated properly. Cases need to be considered where an employee might forget to update the status promptly. For example, the employee might leave his IM status as busy (usual indicator for being in a meeting) even after returning to his cubicle from a meeting.

Conflicts: Conflicts amongst sources providing two different location estimates need to be addressed as well. For

[1]We did not include volunteer-owned cell phones as the volunteers may not be open to installing a tracking application on their private phones. On the other hand, the laptops are supplied and managed by the employers.

Table I
DETAILS OF *Context-Sources* CONSIDERED IN THE PILOT

Context-Source	Context-Cues
Wi-Fi Access Points	RSSI values of Wi-Fi access points seen by employees' laptops were mapped to a numeric scale in the range 0 (no signal) to 100 (very strong signal). 32 Wi-Fi access points are installed in our pilot site and subsets of six or seven of them are reachable at different areas.
System Activity	Provides a binary output specifying the state of the employee's laptop (Idle, Active).
IM Client	Status updates of employees' internal IM client (Available, Away, Busy, In a Meeting, etc.) were recorded during the data collection period.
Calendar	Reports if the employee is scheduled to attend a meeting with a meeting room. Takes a value from the set (NoMeeting, MR204, MR205 etc.)
Time-of-Day	To capture diurnal patterns of employees' activities. Time is represented as an epoch value.

example, consider a case where an employee is attending a meeting in a specific meeting room (obtained from his calendar entry) but leaves his laptop in the cubicle. The Wi-Fi would indicate location as the cubicle whereas the calendar would indicate that the volunteer is in the meeting room.

Heterogeneity of Spaces: Conventional localization systems primarily focus on closed spaces such as a meeting room with a single entry/exit. However, a large portion in commercial sites are often open spaces, like cubicles. In fact, just about 5 percent of our candidate office is closed spaces while the remaining area belongs to open spaces, hall ways etc. We focus on both open and closed spaces. It is challenging to monitor open spaces as they generally have multiple entries/exits and seating densities vary across the area.

In addition, the fundamental nature of information provided by context sources is different for each source. For example, Wi-Fi provides location stamps based on the location of laptops, an IM client refers to the current user activity, whereas time-of-day captures daily patterns in a series of activities an employee engages in. In order to meet the challenges discussed above and to incorporate such dissimilar cues, we model the context of an employee as *Situation*. A situation is an abstraction of an employee's location, activity and behavioral patterns and can be considered as a function of observed context cues. Even though it can be inferred at different granularity levels (e.g, browsing internet, reading a research paper etc.), we are focusing at a level that helps infer area occupancy. Such situations include whether the employee is *at her Cubicle, attending a Meeting, or taking a Break*. As an illustration, the *Break* situation is a categorization for all the activities when the employee is neither attending a meeting (in a meeting room) nor working in her cubicle.

IV. SITUATION INFERENCE

Let possible situations be $S = [S_1, S_2, S_3, ..., S_N]$ and input vector be $C_t = [C_{1t}, C_{2t} \dots C_{Nt}]$, where C_{it} is the cue from the ith context source at time t. Our objective is to infer $S_t \in S$, the situation at time t, from the context cues at that time instant. In the pilot, 36 context cues from five context sources - 1. Wi-Fi (32 context cues from access points[2]), 2. System Activity, 3. IM client, 4. Calendar and 5. Time-of-Day - form the input vector, C. Each training instance relates such an input vector $C_{t'}$ to the manually tagged situation, $S_{t'}$, of the user at time t', where $S_{t'} \in S$. Next we describe two inference models that we used.

A. Regression-based Model

In this model, we consider relative contributions of each context cues to each of the situations and combine them as follows:

$$w_1.C_{1t} + w_2.C_{2t} + \ldots + w_N.C_{Nt} = S_j \qquad (1)$$

where $[w_1, w_2 \dots w_N]$ are the unknown weights assigned to observations $[C_{1t}, C_{2t} \dots C_{Nt}]$ for situation S_j. Before we solve $W.C = S$ using standard techniques to get W, we need to assign numeric values to the observation vector C and the situation vector S.

Computing Observation Vector C : We assign $P(C_{it}|S_j) * idf(C_{it})$ as the numeric value to corresponding elements in the C vector ($i = 1 \dots N$), where $P(C_{it}|S_j)$ is computed from the training data as the Probability Mass Function (PMF) of the observations from context source C_i, for the situation S_j; $idf(C_{it})$ is Inverse Document Frequency (IDF) measure of C_{it}. The IDF measure enables us to de-emphasize same context cue that occur in multiple situations.

Computing Situation Vector S : For S, we transform each S_i to a Cartesian point in (x, y) (say S'_i) plane such that for any other situation S_j, $|S_i - S_j| \geq \Delta$. S'_i sets target point for situation S_i during test time and the value of Δ defines the distance between two target points. We associate a threshold value τ around each S'_i to classify between an 'Inferred' and a 'Not Inferred' state. Essentially, as τ increases more states could be inferred, but, at the cost of precision. Finally, at test time, given an input observation $C(t')$ (context values at time t'), and the W vector (pre-computed using training data), we assign a situation S'_i, if *Euclidean distance*$(S'_i, C) \leq \tau$. For the rest, we mark them *not inferred*.

In our prior paper [9], we describe applying this regression-based approach to building energy management.

B. Classification:

In this approach, user situation is marked as the target variable and a classification tree is constructed using the C4.5 algorithm. Total number of classes is equal to the

[2]Values from unreachable access points are marked as 0 to complete the input vector.

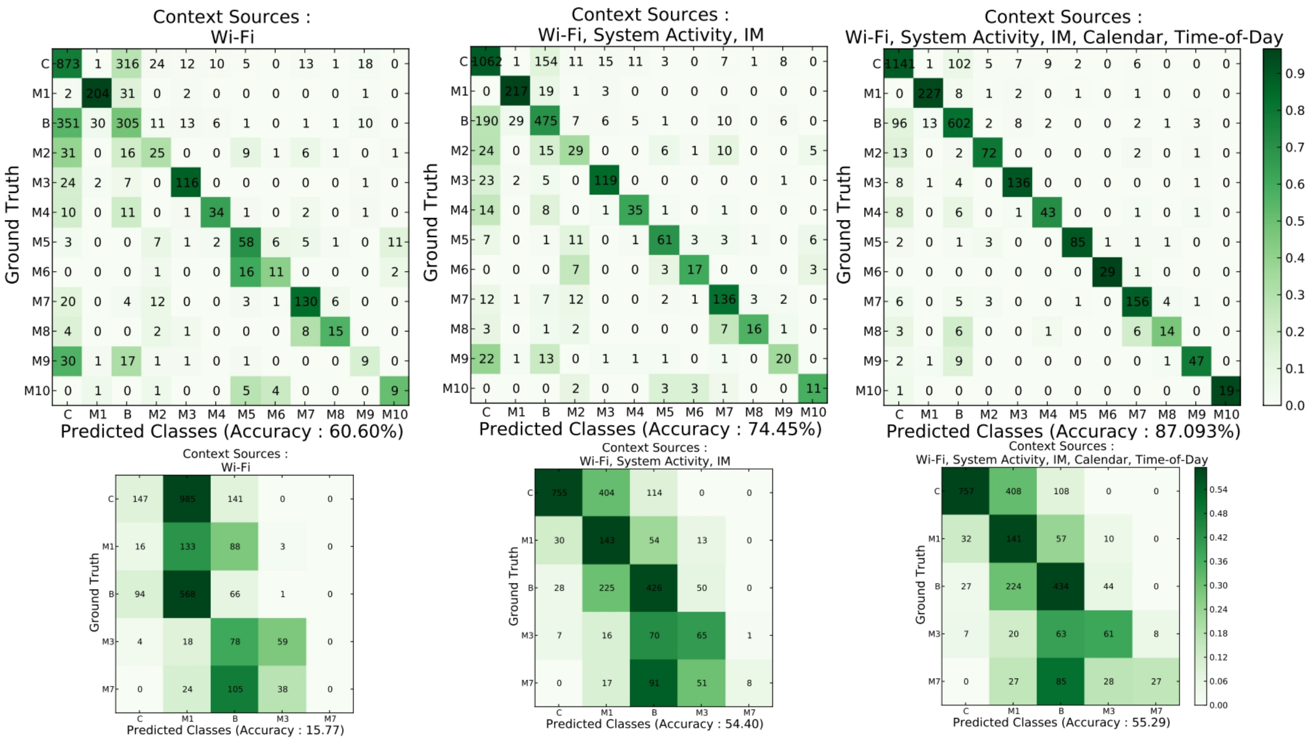

Figure 1. Confusion matrices representing accuracy of inferring one volunteer's situation using classification (top three) and regression (bottom three) approaches using data from subsets of context-sources. Higher numbers in the primary diagonal indicate higher accuracy of inferences. On the other hand, higher numbers in the non-primary diagonal indicate lower accuracy of inferences. It is important to note that both precision and recall for classification based approach is much higher than that for regression-based one.

number of interested situations of a volunteer. Later, each test instance ***I*** is mapped to one element of the set of possible class labels. As shown in Figure 1, the classification approach yields more accurate results than the regression based one. Another benefit of using classification is that it can process both numeric and nominal input variables, therefore, no need of mapping the context cues to numeric values.

V. Results

We use Weka [4] to construct classifier and linear regression models, as well as to evaluate those models using the 10-fold cross validation approach. J48, a Java implementation of the C4.5 algorithm [6] in Weka tool-set is used to construct a well-pruned decision tree in each run that avoids over-fitting. The tool-set also provides a linear regression module, however, pre-processing of the data to convert non-numeric values to numeric values was done externally in Python.

A. System Accuracy

In our evaluation, we quantify system performance based on correctly labelled data instances. While evaluating classification approach we consider all situations, as shown in Figure 1 (top-row). On the other hand for regression, we consider only situations that have sufficient number of tagged data points necessary for effective learning and testing. For instance, as shown in Figure 1 (bottom-row), only five situations have been considered in regression.

Table II
Precision and Recall values for different situations considered for Volunteer 1

Class (Situation)	Precision/Recall for Classification	Precision/Recall for Regression
Cubicle	0.891/0.896	0.919/0.594
Break	0.808/0.826	0.580/0.595
M1	0.934/0.946	0.171/0.585
M3	0.883/0.907	0.426/0.383
M7	0.886/0.886	0.771/0.161

Table II shows precision and recall values of inferred situations for one volunteer that could be considered for both regression and classification approaches. Classification works better even for the situations where sufficient training data is not available. While regression shows high precision values for two situations, recall values for all the situations are poor. For rest of the 4 volunteers, we show average accuracy values obtained in Table III.

Table III
AVERAGE ACCURACY VALUES FOR REST OF THE 4 VOLUNTEERS (OVERALL PERCENTAGE ACCURACY)

Volunteer	Classification	Regression
Volunteer2	89.72%	64.21%
Volunteer3	90.94%	75.60%
Volunteer4	88.66%	80.68%
Volunteer5	87.59%	63.45%

B. *Contribution of individual sources*

In order to highlight the significance of each context source, we start by creating an input vector based on only Wi-Fi access points and then iteratively include context data from other sources. Figure 1 (top-left) shows a *confusion matrix* for predictions made by the classification model while considering only access points. It represents the dispositions of data instances - the numbers along the main diagonal denote the correct decisions made and the numbers out of this diagonal are the errors (confusion) between various classes. It is clear that the model is unable to distinguish between Cubicle and Break (C and B) classes because as explained above the model is not able to distinguish between the location of the volunteer and that of her laptop. Next, in Figure 1 (top-middle), we include System Activity and IM client along with access points. There is a considerable increase in the accuracy of predicting Cubicle and Break situations because these additional context sources can help determine if the volunteer is taking a break or using the laptop in her cubicle. Figure 1 (top-right) considers all the context sources by including Calendar and Time-of-day. These sources increase accuracy of inferring most situations related to the meeting rooms because calendar entries point to locations where meetings are scheduled and time-of-day captures volunteers' temporal regularities in taking breaks, attending meetings, etc.

Similarly, the bottom three matrices in Figure 1 are confusion matrices for regression based approach. Using only access points (bottom-left), the model is confused between three situations and reports a poor accuracy. Addition of IM client status and System Activity improves inference for Cubicle and Break situations. However, the model does not capture patterns in the usual activities of the volunteer using time-of-day parameter.

VI. RELATED WORK

Most commonly used sensors are Passive Infrared (PIR) based sensors, but they can not determine number of people in a given area. CO_2-based occupancy detection sensors [7] are slow to detect change of events; whereas large-scale deployment of Camera [5], [2] or sonar-based surveillance networks [8] incurs substantial deployment costs and maintenance overhead. They also bring up privacy issues as they detect more than what is required. We argue for selective deployment of high precision occupancy sensors only in areas where opportunistic context sources are inadequate.

Erickson et al.[3] propose a more sophisticated wireless network of cameras to determine coarse-grained floor-level occupancies, while maintaining user comfort standards [2]. A more recent work [1], uses low cost and incrementally deployable wireless PIR sensors with reed switches. Our research augments this body of work by advocating a prior context profiling, and smart identification of areas where occupancies can be detected using existing *soft sensor* data.

VII. CONCLUSION AND FUTURE WORK

In this paper, we presented a novel hardware-sensor less scheme that leverages pre-existing opportunistic context sources to derive occupancy details of a commercial building. Each soft sensor provides a cue about an employee's situation, and supervised learning algorithms fuse such cues to infer situations and subsequently to occupancy figures.We report system accuracy of as high as 90% through a pilot study, which demonstrates that this scheme can support context aware applications in smart buildings with minimal or no additional sensors. We are conducting another pilot of similar scale at a different building. In this pilot, along with devising further techniques to increase the system accuracy, we are exploring potential domain specific applications of the scheme, such as dynamic seat allocation. Handling privacy concerns and investigating unsupervised techniques in detail remain focus of our work in the future.

REFERENCES

[1] Y. Agarwal, B. Balaji, R. Gupta, J. Lyles, M. Wei, and T. Weng. Occupancy-driven energy management for smart building automation. BuildSys '10, pages 1–6, New York, NY, USA. ACM.

[2] V. L. Erickson and A. E. Cerpa. Occupancy based demand response hvac control strategy. BuildSys '10, pages 7–12, New York, NY, USA, 2010. ACM.

[3] E. et all. Energy efficient building environment control strategies using real-time occupancy measurements. BuildSys '09, pages 19–24, New York, NY, USA, 2009. ACM.

[4] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*

[5] A. Kamthe, L. Jiang, M. Dudys, and A. Cerpa. Scopes: Smart cameras object position estimation system. EWSN '09, pages 279–295, Berlin, Heidelberg, 2009. Springer-Verlag.

[6] J. R. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[7] Review of CO2 demand controlled ventilation technology and application. U.S. Dept. of Commerce, National Institute of Standards and Technology. Gaithersburg, Md. : 2001.

[8] S. P. Tarzia, R. P. Dick, P. A. Dinda, and G. Memik. Sonar-based measurement of user presence and attention. Ubicomp '09, pages 89–92, NY, USA, 2009. ACM.

[9] L. Thanayankizil, S. Ghai, D. Chakraborty, and D. Seetharam. Softgreen : Towards energy management of green office buildings with soft sensors. In *Energy in Communication, Information, and Cyber-physical Systems (E6) Workshop, COMSNETS*. IEEE, 2012.