Occupancy Detection Through Light, Humidity, Carbon Dioxide and Temperature Measurements

Yerbol Aussat
Department of Computer Science
University of Waterloo
Waterloo, Canada
yaussat@uwaterloo.ca

***Abstract*: Accurate determination of office occupancy is an important problem, the solution to which could enable many compelling applications. Heating, ventilation and air conditioning (HVAC) as well as lighting are two major energy consumers in buildings. Previous studies showed that implementing demand driven lighting and HVAC operations could substantially reduce building energy consumption. However, such solutions rely on accurate occupancy information. This study evaluates the accuracy of the prediction of occupancy in an office room using data from temperature, light, humidity and carbon dioxide sensor measurements with different statistical classification methods using the open source Python Scikit-Learn library. In the literature, these statistical methods are also referred to as machine learning-based multi-sensors data fusion methods [11]. All models have been tested on three datasets (one training and two testing datasets), and their performance has been evaluated using the conflict matrix. Typically the best accuracies are obtained from training k-Nearest Neighbors (k-NN), Decision Trees (DT) and Random Forrest (RF) models. When all of the features are taken into consideration, the average testing accuracy of 99.4% is achieved by the random forest model. Light has proved to be the best single occupancy predictor. When only carbon dioxide level, humidity, week status (WS) and number of seconds from midnight (SM) are taken into consideration, the random forest model achieves the average testing accuracy of 99.04%. Also, it was shown that using the features extracted from the time stamp, SM and WS, significantly improves prediction accuracy when decision trees or random forest models are used.**

Keywords—Building energy consumption, Occupancy detection, Occupancy estimation, HVAC, Intelligent lighting, Non-intrusive sensor, Data mining, Data fusion, Artificial Neural Networks, Decision Trees, Random Forest, Naïve Bayes, k-Nearest Neighbors, Support Vector Machines, Machine Learning

1. INTRODUCTION

The ability to accurately determine a room's occupancy makes a number of compelling applications possible. According to a recent literature review, commercial buildings in the United Sates account for about 41% of national energy consumption [4]. 39.6% of the total energy consumption in commercial buildings was consumed by heating, ventilation and air conditioning (HVAC) systems, and 20.02% - by lighting systems, both of which are essential for commercial buildings comfort.

Many researchers claim that determining occupancy in the room is a key for reducing the energy consumption in commercial buildings. A recent literature review shows that with the accurate determination of occupancy around 20-50% lighting and HVAC energy can be saved. For instance, the study by Brooks et al. demonstrates that it is possible to reach 37% of HVAC energy savings by taking into consideration the occupancy of a building [6]. In another study by Brooks et al. between 29% and 80% of energy savings were reached when the occupancy information was used as an input to HVAC control algorithms [7]. Similarly, Liu et al in their study reported that the lighting energy savings of 30% could be achieved when occupancy data is taken into consideration in lighting control [8].

Occupancy information can also be valuable for security applications as well as for prediction of occupant behavior.

Nowadays common systems for occupancy detection are based on passive infrared sensors (PIR) or digital cameras. Using PIR sensors alone is a not always a reliable method because such systems fail to detect occupancy accurately when an occupant doesn't move much for a certain amount of time. Also, PIR sensors get easily triggered by air currents, which again leads to inaccurate occupancy detection. On the other hand, the solutions based on digital cameras cause privacy concerns. For instance, employees might not feel comfortable being under constant digital camera surveillance in an office, which might negatively affect their productivity

In this research project a combination of measurements from temperature, carbon dioxide, light and humidity sensors was used to detect occupancy. The records from a digital camera were used to establish ground occupancy for supervised classification model training. Different combinations of these sensors are already available in many modern buildings. The naïve Bayes, decision tree learning, random forest, support vector machines, k-nearest neighbors and artificial neural network models were used in this work. All of these statistical classification models were obtained from open source Scikit learn Python library.

This work uses findings of previous research stating that better occupancy detection can be achieved by using monitoring equipment with higher resolution and accuracy. Also, it was claimed that using a combination of different sensors for the occupancy detection problem tends to demonstrate better results [1].

1.1 Prior work

Occupancy detection is an open research problem, and many research groups propose different solutions for this problem. Thus, in their paper Quian et al propose a Wi-Fi-based occupancy sensing system. The system is able to estimate human's location at a decimeter level. Unlike previous systems that used statistical learning techniques and required training, Widar uses a geometrical model that captures the relationships between Channel State Information (CSI) signals and human's velocity / location. The system is capable of simultaneously and directly tracking both velocity and location of a person, which provides us rich mobility information for various applications. The Widar system was implemented and its performance was validated on COTS Wi-Fi hardware [8].

A different approach to the occupancy detection problem was examined by Candanedo et al, who used measurements from light, temperature, humidity and carbon dioxide sensors to determine occupancy of an office. In order to tackle the problem they used Linear Discriminant Analysis (LDA), Gradient Boosting Machines (GBM), Classification and Regression Trees (CART), and Random Forrest (RF) models. The best accuracy rates of 95-99% were achieved with LDA, CART and RF methods. The results of their work show that a proper selection of features and classification model can have an important impact on the occupancy prediction [1]. The same dataset was used in our study.

A similar approach was used by Tutuncu et al. They applied a few different Artificial Neural Network-based algorithms to the dataset containing measurements of carbon dioxide, temperature, humidity and light sensors. As a result of their study, the Limited Memory Quasi-Newton algorithm demonstrated the highest performance with the accuracy rate of 99.061%. The lowest accuracy rate of 80.324% was obtained by Batch Black algorithm [10].

Hailemariam also used different types of sensors in tandem to determine single person office occupancy in real time. Their work is utilizing decision tree model to explore different features derived from sensor measurements and their importance for the occupancy determination problem. Combinations of measurements from carbon dioxide, current, light, sound and passive infrared motion sensors are used in this work to predict occupancy. The authors found that the best predictor of occupancy is the root mean square error of a passive infrared motion sensor, calculated over a two-minute period. The accuracy rate obtained by using this feature alone is 97.9%. By combining several features derived from the motion sensor, the accuracy rate becomes 98.4%. Using other additional features from sound, carbon dioxide, current and light sensors worsens the classification results. According to the authors, the most likely reason of declining accuracy with adding more features is overfitting, to which decision trees are generally quite vulnerable [2].

Dong and Andrews approached the problem of saving energy in buildings by learning occupancy behavioral patterns of occupants instead of predicting the occupancy directly from sensor measurements. They developed and implemented algorithms for sensor-based modeling and prediction of user behavior in intelligent buildings and connected behavioral patterns to building comfort management systems through simulation tools [5]. Similarly, Tina Yu proposed a genetic algorithms - based model for occupancy behavior for energy efficiency and occupants comfort management in intelligent buildings. As a result of her work, 80-83% accuracy on testing data was achieved. As the author pointed out, predictive occupancy behavior model, based on the historical data, is particularly useful for more accurate energy consumption estimations [13].

2. DATASET DESCRIPTION

The dataset used in this study is the "Occupancy detection data set" from the UCI Machine Learning Repository. The original study, during which the dataset was generated, is [1]. The dataset can be accessed through the following link:
https://archive.ics.uci.edu/ml/datasets/Occupancy+Detection+

Temperature, *relative humidity*, *light* and *carbon dioxide* sensors were continuously recording the data over the period of two weeks. The experiment took place in the month of February. The sensors were placed next to the workstation in an office room with the dimensions of 5.85m x 3.50m x 3.53m (W x D x H). The description of the used sensors is shown in Table 1. A digital camera was used to determine the occupancy status of the office. Images from the camera were recorded every minute. Later these images were used to manually label the data with the occupancy status, establishing the ground occupancy. The sensors were controlled by Arduino microcontroller, and sensor measurement were collected once every 14 seconds, or 3-4 time per minute. The digital camera was controlled by Raspberry Pi microcontroller [1].

Sensor	Parameter	Accuracy	Resolution
DHT22	Temperature (T)	±0.5℃	0.1℃
DHT22	Humidity (φ)	±3% RH	0.1%
TSL2561	Light (L)	NA	1 Lux
Telaire 6613	CO_2	±30 ppm	1 ppm

Table 1: Monitoring equipment

The labeled occupancy value is a binary value equal to 0 and 1 for non-occupied and occupied statuses respectively.

2.1. Data Pre-processing and feature extraction

In addition to the row measurements from the sensors, several higher-level features have been used in the data model. One of the features is a humidity ratio, which is a complex value derived from temperature and relative humidity. The humidity ratio, *W*, is calculated by the following expression:

$$W = 0.622\frac{p_w}{p - p_w}$$

where p is a standard atmospheric pressure (100.326 kPa), and p_w is the saturation pressure over liquid water:

$$p_w = \varphi * p_{ws}$$

$$\ln(p_{ws}) = \frac{C_1}{T} + C_2 + C_3T + C_4T^2 + C_5T^3 + C_6\ln(T)$$

where coefficients C_i are tabular constants.

In addition to humidity ratio, two features were extracted from the time stamp: *Seconds from Midnight (SM),* and *Week Status (WS)*. *SM* is a number that uniquely represents time of the day, and is calculated by the following equation:

$$SM = 3600 * hour + 60 * minute + second$$

Week Status, WS, is a binary value that equals 0 if the day is a workday, and 1, if the day is a weekend.

2.2. Preliminary analysis of individual signals

Figures 1-6 show individual signals from each of the sensors (carbon dioxide, light, humidity levels and temperature) as well as the signals for humidity ratio and ground ("true") occupancy. From the plots it can be clearly seen that there are patterns in each of the signals. Vertical lines on the signal plots correspond to the midnight.

From the CO_2 plot on Figure 1 it can seen that at midnight the carbon dioxide level in the office is low. Then, around 8-9am it gradually grows and stays high during the working hours. After the end of the working hours, namely, after

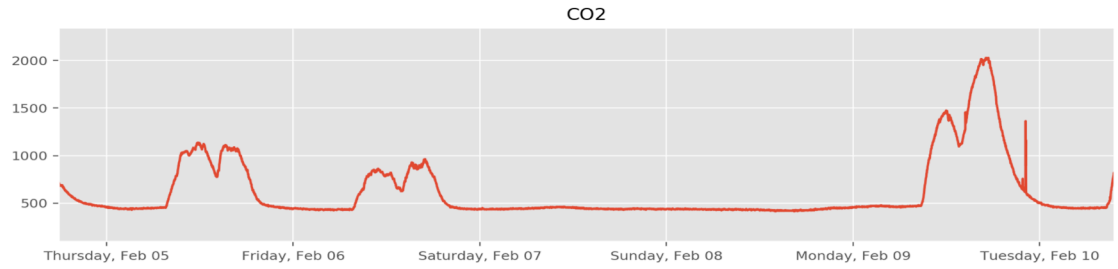

Fig. 1. Signal for CO_2 sensor

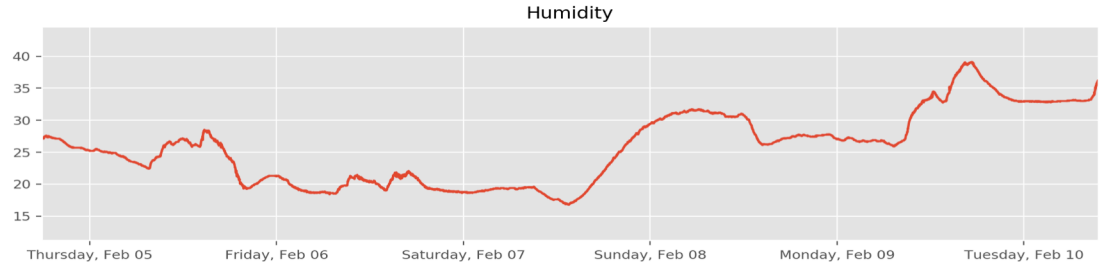

Fig. 2. Signal for humidity sensor

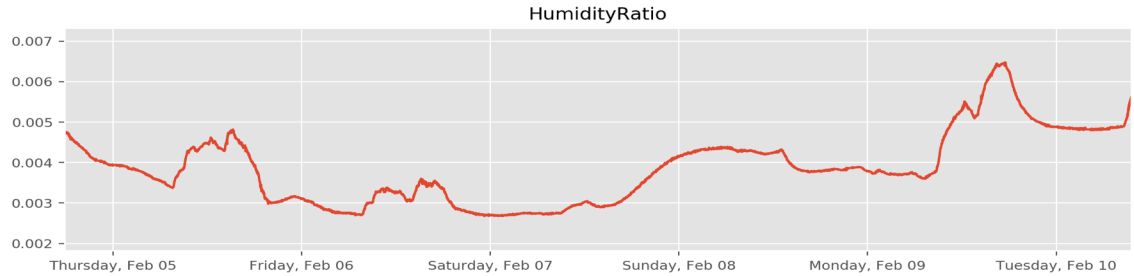

Fig. 3. Signal for humidity ratio

~6pm, the level of carbon dioxide gradually decreases back to the low level value. It should be noted that for all workdays depicted on the Figure 1, in the middle of the day there is a fluctuation in the CO_2 level, which likely corresponds to a lunch break. Also, the carbon dioxide level stayed low over the weekend, implying that the room wasn't occupied.

From the signals for humidity and humidity ratio, illustrated on Figures 2 and 3, it can be noticed that there are fewer fluctuations on the plot over the weekends. In general, it is hard to point out any clear patterns just by visually inspecting these plots. The two plots have very similar shape, from which a strong dependence of humidity ratio on humidity can be guessed.

Figure 4 illustrates the time series for the light sensor. As it can be seen from the plot, at the moment when the light is turned on, there are abrupt jumps in light signal. The signal is a combination of artificial lighting and daylight contribution. Thus, on the weekend smooth light changes due to daylight contribution can be seen. During the workdays, one can clearly observe a superposition of daylight and artificial lighting. Also, similarly to the carbon dioxide signal, light signal fluctuation can be seen in the middle of workdays, which likely corresponds to the occupant taking a lunch break.

Temperature signal is illustrated on Figure 5. Again, a repetitive pattern can be noticed on the plot. Overall, temperature is high during daytime, and low during nighttime. Unlike light levels, which more or less have the same high and low values, temperatures don't have constant high and low values. Some days and nights can be warmer, and others can be colder. However during the work hours the temperature seems to be maintained at a certain comfortable level, which is 22-23℃. Another observation is that there seem to be more temperature fluctuations during workdays, compared to the weekends, when temperature changes smoothly without any abrupt transitions.

The last figure illustrates ground occupancy. As we can see, the office was not occupied on the weekend. In general, since the occupancy data was recorded in an office, there is a

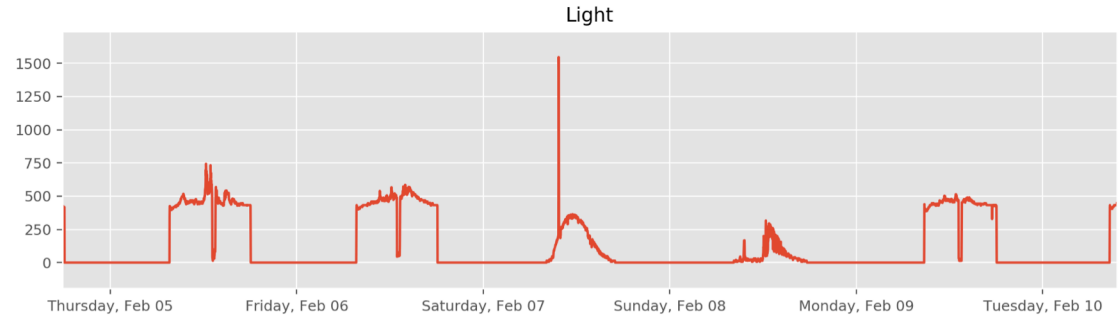

Fig. 4. Signal for light sensor

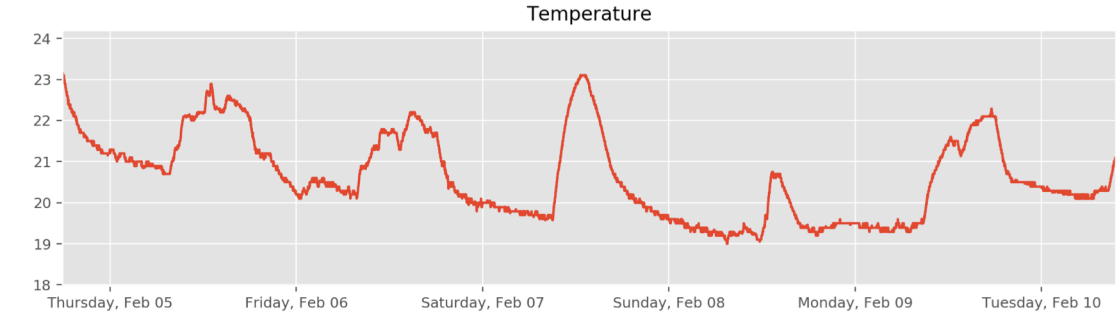

Fig. 5. Signal for temperature sensor

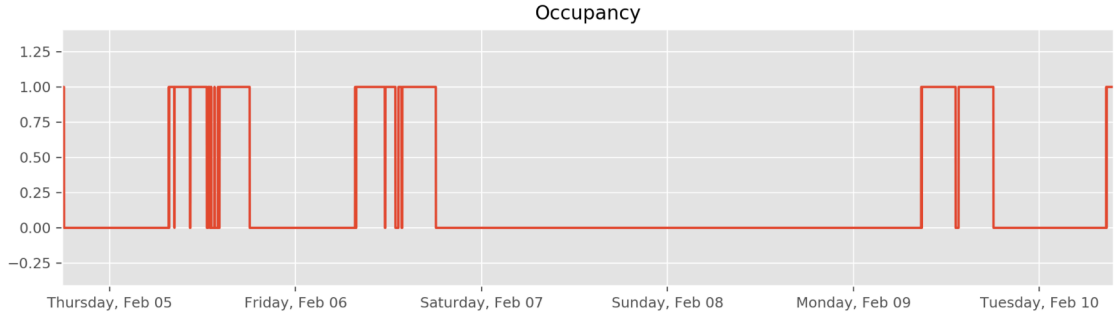

Fig. 6. Signal for ground occupancy

clear occupancy pattern on the workdays. It looks like the occupant comes, leaves and takes a lunch break roughly at the same time every day.

Figure 7 illustrates average scaled values of temperature, humidity, light and carbon dioxide levels for occupied and unoccupied cases. The values are scaled, according to the following equation:

$$X_{Scaled} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

From the bar diagram it can be seen that average values of temperature, light and carbon dioxide are significantly higher when the office is occupied, whereas the average value of humidity is only slightly higher for the occupied status. It should be pointed out that the bar diagram doesn't imply direct correlation between occupancy status and each of the parameters. For example, according to the diagram, temperature and light levels are much higher when the room is occupied. However, it is mostly due to the fact that temperature and light levels tend to be higher during the day, and, at the same time, offices are typically occupied during the day.

3. Methodology

3.1. Training and testing data sets

The original dataset has been split into 3 datasets. The first dataset is a training dataset, on which the classification models have been trained. Two other datasets are testing datasets, which were not seen by the algorithms during the training time. These datasets are intended for evaluating the performance of each algorithm. The fraction of data points with the occupancy status of 1 is roughly 23.1% in each of three datasets. The detailed description of each dataset is shown in Table 2.

Data Set	Number of Samples	Percent of total samples
Training	14391	70%
Testing 1	3085	15%
Testing 2	3084	15%
Total	20560	100%

Table 2: Training and Testing Datasets

3.2. *Techniques*

According to F. Alam et al, data fusion techniques can be divided into three categories based on the mathematical underlying methods [11]:

- **Probability-based techniques** including Bayes' theorem based methods, statistics, and recursive operators.
- **Artificial Intelligence-based techniques** including supervised machine learning algorithms, fuzzy logic, artificial neural networks (ANN) and genetic evaluation.
- **Theory of Evidence-based** techniques including Dempster-Shafer theory

In order to approach the problem of occupancy detection, six models have been used: naïve Bayes, decision tree learning, random forest, support vector machines, k-nearest neighbors and artificial neural network. In this section a brief description for each of the techniques will be provided.

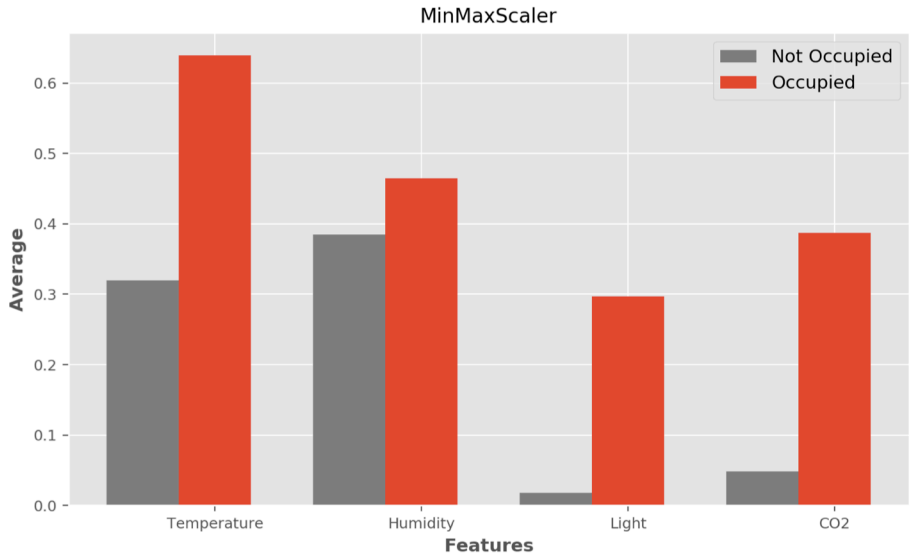

Fig. 7. Scaled average individual signals for Occupied and Unoccupied statuses

3.2.1. Naive Bayes (NB)

Naïve Bayes is a probabilistic classifier, which is based on applying Bayes' theorem with strong (naïve) independence assumption between the features. Using Bayes' Theorem the conditional probability can be expressed in the following form:

$$P(C_k|\boldsymbol{x}) = \frac{P(C_k)P(\boldsymbol{x}|C_k)}{P(\boldsymbol{x})}$$

where C_k is a category (label) and $\boldsymbol{x}$ is a set of evidences. In plain English, using Bayesian probability terminology, the Bayes' theorem can be written as:

$$Posterior = \frac{Prior * Likelihood}{Evidence}$$

Under the naïve independence assumptions, the category (label) can be determined as:

$$\hat{y} = argmax\, P(C_k) \prod_{i=1}^{n} p(x_i|C_k)$$

In many practical applications, parameter estimation for naïve Bayes models uses the method of maximum likelihood [12]. Despite their oversimplified assumptions, Naïve Bayes classifiers perform well in many real-world situations. F. Alam et al categorize Naïve Bayes as a probabilistic data fusion algorithm [11].

In this work Gaussian naïve Bayes was used, which is suitable for dealing with continuous data. The typical assumption is that the continuous values associated with each class are distributed according to Gaussian distribution.

3.2.2. Decision Trees (DT)

Decision tree learning uses a decision tree to go from observations about an item to conclusions about the item's target value. It is one of the predictive modeling approaches widely used in statistics, data mining and machine learning. Leaves of the decision tree represent labels, nodes – features, and branches – values of features. Decision tree models where the target variables take a discrete set of values are called classification trees [12]. DT is a machine learning-based multi-sensor data fusion method [11].

In this work, in order to avoid overfitting, maximum depth of the decision tree was set to 10.

3.2.3. Random Forrest (RF)

Random forest, or random decision forest is an ensemble learning method. It operates by constructing many decision trees at training time. That's where the name "forest" comes from. At testing time, it outputs the class that is the mode of the classes predicted by single decision trees. Random forest alleviates the decision tree model's tendency to overfit [12].

3.2.4. Support Vector Machines (SVM)

Support vector machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. SVM is also considered a machine learning-based multi-sensor data fusion method [11]. It is a non-probabilistic binary linear classifier. SVM model represents all examples as points in n-dimensional space, where each of the dimensions corresponds to a feature. SVM then maps these examples in such a way that examples of different categories are divided by a clear gap that is as wide as possible. At the prediction time, new samples are mapped into the same space, and their categories are determined based on which side of the gap they fall [12].

3.2.5. *k-Nearest Neighbors (k-NN)*

k-Nearest Neighbors algorithm is a non-parametric method used for classification and regression. The input of k-NN is a set of k closest training examples in the feature space, and, for a classification problem, the output is a class. The object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. k-NN is a type of instance-based learning or lazy learning, where the function is only approximated locally and all computation is deferred until classification [12]. It is among the simplest machine learning algorithms. k-NN is a machine learning-based multi-sensor data fusion method [11]. In this work, the number of nearest neighbors k was set to 3.

3.2.6. Artificial Neural Network (ANN)

Artificial Neural Networks have emerged as the simulation of the biological nervous system. ANN is composed of neurons, which can be connected to each other in a very complex way. Each connection (synapse) between neurons can transmit a signal to another neuron. The receiving neuron can process a signal, and then send a resulting signal to the downstream neurons connected to it. Neurons and synapses have a weight that varies as learning proceeds, which can increase or decrease the strength of the signal that it sends downstream. The neurons also might have a threshold such that only the aggregate input signal above certain strength produces an output signal. Neurons are typically arranged in layers, and signals travel from the input layer to the output layer [12]. ANNs have an extraordinary ability to extract patterns in highly complex data sets. F. Alam et al categorize Artificial Neural Networks as artificial intelligence based data fusion algorithms. Fusing historical data by training and testing [11]. For the purposes of this work, a neural network with 2 hidden layers of 30 neurons each was used. The learning rate was set to 0.001.

3.3. Training and testing data sets

A confusion matrix contains information about actual and predicted classifications done by a classification technique. Performance of each of the algorithms can be evaluated using the data in the matrix. A confusion matrix for two-class classifier is shown in Table 3. The matrix has following entries:

a: number of correct predictions that an instance is negative

b: number of incorrect predictions that an instance is positive

c: number of incorrect predictions that an instance is negative

d: number of correct predictions that an instance is positive

		Predicted	
		Negative	Positive
Actual	Negative	a	b
	Positive	c	d

Table 3: Confusion Matrix

Given the accuracy matrix, the accuracy rate, or the correctly classified ration, can be determined using the equation below. Accuracy rate is the proportion of the total number of predictions that were correct.

$$\%CCR = \frac{a + d}{a + b + c + d} * 100$$

We will use this accuracy rate to evaluate the performance of our techniques.

4. RESULTS

4.1. Simulation results

In order to tackle the occupancy detection problem six statistical models were used. They are naïve Bayes (NB), decision trees (DT), random forest (RT), support vector machines (SVM), k-nearest neighbors (k-NN) and artificial neural networks (ANN).

Each of the models has been trained on the training data set consisting of 70% of all data points. Next, the models have been evaluated against the training, testing 1 and testing 2 data sets. Confusion matrix is used to evaluate the prediction accuracy of each model. As it was shown earlier, the accuracy of each model is calculated by the sum of true positives and true negatives, divided by the total number of predictions. Training of the models was done using the open-source Python Scikit-learn machine-learning library.

All models have been trained using different combinations of input features, also known as predictors. The results are summarized in Table 4.

4.2. Discussion

This section compares the results from Table 4, which have been achieved by different prediction models as well as different combinations of predictors.

High accuracies on both training and testing data sets have been achieved for all models except for support vector machines when all the measured parameters are taken into consideration for model training. The accuracy rates for the testing sets for naïve Bayes, decision trees, random forest, k-nearest neighbors and artificial neural networks are all around 97-99%. On the other hand, SVM model shows 100% accuracy on the training set, but only 75-76% accuracy on the testing sets. Therefore, it can be concluded that SVM overfits, when all of the predictors are used for training. Likely, the reason of overfitting is heterogeneity of the input features.

In the next set of experiments, the models have been trained on all of the parameters, except for the ones derived from the time stamp: seconds from midnight (SM) and weekday status (WS). Excluding these two predictors makes the data less heterogeneous. As a result, all of the algorithms, including SVM, have done well on testing datasets, with the accuracy rates ranging from 95% to 99%.

When only lighting and carbon dioxide level measurements are available for training, all of the algorithms perform quite well with decision trees, random forests, and k-nearest neighbors models achieving the best accuracies of 98% and higher. The worst performance is demonstrated by SVM, which had accurately determined occupancy in only 93% of all tests. When only temperature and carbon dioxide level measurements are available, the performance of all algorithms worsens. Random forests and decision trees show the best accuracy rates of ~94% for both testing sets. Naïve Bayes and artificial neural networks perform poorly with only 82% and 76% of average accuracy respectively. From these results, it can be guessed that the light is a stronger predictor of the occupancy than temperature. If relative humidity measurements are considered in addition to temperature and carbon dioxide levels, the accuracy rate increases to 97.8% when using the random forest model. Finally, as we know, humidity ratio is a feature that is derived from temperature and relative humidity, and therefore, it can be used when both temperature and humidity measurements are available. Thus, if humidity ratio, relative humidity, temperature and carbon dioxide levels are used as features, 98.3% accuracy can be achieved by random forest model. As we can see from Table 4, both random forest and decision trees models perform better when more features are available. It is likely due to the underlying mechanisms of these two algorithms. During the construction of classification trees, both algorithms select the

features that maximize information gain. Then the higher importance is assigned to the early selected features, and lower importance to the features that were selected late.

Next, in order to determine the effect of each individual parameter on the prediction accuracy, a series of tests has been conducted with only one single parameter taken into consideration. As it can be seen from Table 4, lighting level is a single strongest predictor of occupancy with the average accuracy rate of 98.7% achieved on two datasets by the artificial neural networks model. Decision trees, random forests, and k-nearest neighbors models also demonstrate very good accuracy, producing the results close to ANN. The rest of the features, when used alone, don't show good prediction accuracies. The accuracy rates for the cases when temperature, humidity ratio and carbon dioxide levels are considered individually are between 79% and 85%.

Model	**Parameters**	**Training Accuracy (%)**	**Testing Accuracy 1 (%)**	**Testing Accuracy 2 (%)**
NB	T, φ, L, CO_2, W, WS, SM	96.685	97.083	96.887
DT	T, φ, L, CO_2, W, WS, SM	99.660	99.254	99.125
RF	T, φ, L, CO_2, W, WS, SM	99.958	99.514	99.319
SVM	T, φ, L, CO_2, W, WS, SM	100.000	75.656	76.978
k-NN	T, φ, L, CO_2, W, WS, SM	99.611	99.125	98.833
ANN	T, φ, L, CO_2, W, WS, SM	96.463	97.018	96.077
NB	T, φ, L, CO_2, W	96.595	96.823	96.984
DT	T, φ, L, CO_2, W	99.514	99.157	98.995
RF	T, φ, L, CO_2, W	99.944	99.319	99.287
SVM	T, φ, L, CO_2, W	99.951	95.170	95.233
k-NN	T, φ, L, CO_2, W	99.305	98.865	98.865
ANN	T, φ, L, CO_2, W	98.694	98.606	98.703
NB	L, CO_2	97.561	96.888	96.952
DT	L, CO_2	99.555	98.639	98.346
RF	L, CO_2	99.910	98.703	98.508
SVM	L, CO_2	99.917	93.225	92.899
k-NN	L, CO_2	99.354	98.865	98.476
ANN	L, CO_2	97.005	96.823	96.628
NB	T, CO_2	82.204	82.950	83.366
DT	T, CO_2	95.233	94.295	93.807
RF	T, CO_2	99.229	94.392	94.844
SVM	T, CO_2	92.989	91.118	91.051
k-NN	T, CO_2	93.746	90.340	90.694
ANN	T, CO_2	76.048	76.207	77.335
NB	T, φ, CO_2	83.483	83.079	82.977
DT	T, φ, CO_2	97.853	96.726	96.239
RF	T, φ, CO_2	99.868	97.958	97.730
SVM	T, φ, CO_2	96.491	92.512	91.375
k-NN	T, φ, CO_2	95.191	91.669	91.051
ANN	T, φ, CO_2	77.611	76.402	77.821
NB	T, φ, CO_2, W	82.864	83.857	83.982
DT	T, φ, CO_2, W	97.589	96.596	95.850
RF	T, φ, CO_2, W	99.910	98.379	98.281
SVM	T, φ, CO_2, W	95.956	92.577	92.380
k-NN	T, φ, CO_2, W	95.330	91.540	91.472
ANN	T, φ, CO_2, W	82.684	83.209	83.625
Best (ANN ~ DT ~ RF~ SVM~ k-NN)	L	98.791	98.639	98.768
Best (SVM ~ DT ~ RF)	T	85.692	84.959	84.857
Best (SVM ~DT~RD~K-NN)	CO_2	88.215	86.321	85.830
Best (RF ~ DT)	φ	85.220	80.130	79.150
Best (RF ~ DT ~ k-NN ~ ANN ~ NB)	L, WS, SM	99.785	99.060	98.768
Best (DT ~ RF)	T, WS, SM	97.401	97.374	96.790
Best (RF ~ DT)	CO_2, WS, SM	99.903	98.509	98.249
Best (RF ~ DT)	φ, WS, SM	99.917	98.574	98.865

Table 4: Performance of 6 techniques on different combinations of predictors

Model	Parameters	Training Accuracy (%)	Testing Accuracy 1 (%)	Testing Accuracy 2 (%)
NB	φ, CO_2	80.773	80.130	79.345
DT	φ, CO_2	95.247	92.577	92.996
RF	φ, CO_2	99.521	94.911	95.493
SVM	φ, CO_2	95.025	89.919	90.110
k-NN	φ, CO_2	94.462	88.979	89.235
ANN	φ, CO_2	83.893	82.853	82.944
NB	φ, CO_2, WS, SM	85.713	85.705	85.636
DT	φ, CO_2, WS, SM	99.194	98.023	98.703
RF	φ, CO_2, WS, SM	99.951	98.865	99.222
SVM	φ, CO_2, WS, SM	99.979	77.439	76.816
k-NN	φ, CO_2, WS, SM	97.901	95.203	94.877
ANN	φ, CO_2, WS, SM	67.487	68.817	68.126

Table 4 (continued)

Interestingly, when the features extracted from the timestamp, WS and SM, are taken into consideration in addition to individual sensor measurements, significantly higher accuracy rates are achieved. Specifically, as it can be seen from Table 4, random forest and decision trees models achieve the average testing accuracies of 97%, 98.4% and 98.7% with temperature, carbon dioxide level, and relative humidity features respectively. This is a very significant improvement. In commercial buildings and offices, where occupancy behavior has strong repetitive patterns, WS and SM can be used in addition to other predictors to enhance the performance. The accuracy of the prediction using light sensor measurements only also slightly improved to 98.9%, when timestamp-based features are taken into consideration.

It should be emphasized that one of the main applications that motivates the office occupancy detection studies is to use this information for reducing energy consumption by intelligent lighting and temperature control. For such applications light levels and temperature cannot be used as predictors for the occupancy. Keeping this in mind, these parameters are not taken into consideration in the next set of experiments.

As it can be seen from Table 4, when only relative humidity and carbon dioxide levels are taken into consideration, the best average testing accuracy of 95.2% is achieved by random forest model. Decision trees, k-nearest neighbors and support vector machines models demonstrate slightly lower prediction accuracies, and artificial neural networks and naïve Bayes techniques perform poorly. When timestamp-based features, WS and SM, are taken into consideration in addition to relative humidity and carbon dioxide level measurements, random forest and decision trees models demonstrate a very strong performance with the average testing accuracies of 99.04% and 98.36% respectively.

Overall, it can be seen that random forest model consistently demonstrates the best performance for the majority of the feature combinations. It can be explained by several reasons. Firstly, random forest requires almost no input preparation, and therefore it can handle highly heterogeneous features. It also performs implicit feature selection according to maximization of information gain. Finally it is less vulnerable to overfitting than decision trees.

5. CONCLUSION

This study demonstrates that it is possible to accurately determine office occupancy by using temperature, light, CO_2 and relative humidity measurements. The average accuracy of 99.4% has been achieved by random forest model when all parameters are taken into consideration. It has been found that support vector machine model doesn't perform well, when the week status and seconds from midnight are added as features. This is very likely due to the fact that the accuracy of SVM declines when the features are highly heterogeneous.

In this work it has been shown that taking into consideration the timestamp-based features, namely, seconds from midnight and week status, substantially increases the accuracy rate of many algorithms. In particular, decision trees and random forest models perform significantly better when these two features are used in addition to others. Thus, when carbon dioxide level, WS and SM are used as predictors, the occupancy can be determined with the accuracy of 98.4% by the random forest model. Similarly, when WS and SM are used in addition to relative humidity, random forest model achieves 98.7% testing accuracy. Using WS and SM is more suitable in an office setting, when occupancy behavior has clear repetitive patterns.

Light turns out to be the strongest single predictor of the occupancy. Random forest algorithm achieves 98.7% accuracy by using only the light level as a single feature. However, in the situations, when the occupancy information is required as an input to smart lighting and temperature control systems, light and temperature measurements cannot be used as occupancy predictors. It has been shown that the accuracy of 99.04% can be achieved by random forest algorithm when

relative humidity, carbon dioxide level, week status and number of seconds from midnight are used as features. It can be concluded that this solution can be effectively used for occupancy detection for smart lighting and temperature control systems.

It should be noticed that the time horizon of the collected data is only two weeks. In the future work, it would be interesting to examine if the performance of the models could be improved further by training them on more data, collected over longer periods of time. The seasonality might be another factor that can affect the results. The data was collected in February, and there are certain ranges of temperatures, relative humilities and light levels associated with this season. Therefore, the same models might not work as good in other seasons. In the future work, the effect of seasonality needs to be examined. It is also interesting to examine how the predictive models would perform at different locations with different climate conditions.

ACKNOWLEDGMENT

This study used the data set from UCI Data Repository. The original authors collected this dataset are Luis M. Candanedo and Veronique Feldheim from University of Mons, Belgium [1].

REFERENCES

[1] Candanedo L.M., Feldheim V., Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models, Energy and Buildings 112 (2016) 28–39

[2] E. Hailemariam, R. Goldstein, R. Attar, A. Khan, Real-time occupancy detection using decision trees with multiple sensor types, in: Proceedings of the 2011 Symposium on Simulation for Architecture and Urban Design, Society for Computer Simulation International, San Diego, CA, 2011, pp. 141–148.

[3] Z. Yang, N. Li, B. Becerik-Gerber, M. Orosz, A multi-sensor based occupancy estimation model for supporting demand driven HVAC operations, in: Proceedings of the 2012 Symposium on Simulation for Architecture and Urban Design, Society for Computer Simulation International, San Diego, CA, USA, 2012, pp. 49–56.

[4] Shen W., Newsham G, Implicit Occupancy Detection for Energy Conversation in Commercial Buildings: A Survey, Submitted to CSCWD 2016,2016

[5] Dong B., Andrews B., (2009). Sensor-based occupancy behavioral pattern recognition for energy and comfort management in intelligent buildings. Proceedings of Building Simulation

[6] J. Brooks, S. Goyal, R. Subramany, Y. Lin, T. Middelkoop, L. Arpan, L. Carloni, P. Barooah, An experimental investigation of occupancy-based energy-efficient control of commercial building indoor climate, in: Proceeding of the IEEE 53rd Annual Conference on, IEEE, Decision and Control (CDC), Los Angeles, CA, 2014, pp. 5680–5685.

[7] J. Brooks, S. Kumar, S. Goyal, R. Subramany, P. Barooah, Energy-efficient control of under-actuated HVAC zones in commercial buildings, Energy Build. 93 (2015) 160–168.

[8] Liu, J., et al., Fuzzy logic controller for energy savings in a smart LED lighting system considering lighting comfort and daylight, Energy and Buildings, 2016. 127: p. 95-104.

[9] K. Qian, C. Wu, Z. Yang, C. Yang, Yunhao Liu: "Decimeter Level Passive Tracking with WiFi" in HotWireless '16 Proceedings of the 3rd Workshop on Hot Topics in Wireless (2016)

[10] Tutuncu K., et al. Occupancy Detection Through Light, Temperature, Humidity and CO2 Sensors Using ANN., Proceedings of the ISER 45th International Conference, Rabat, Morocco, 2016

[11] Alam F. et al. Data Fusion and IoT for Smart Ubiquitous Environments: A Survey., IEEE Access, Volume 5, 2017

[12] Narasimha Murty, M.; Susheela Devi, V. (2011). Pattern Recognition: An Algorithmic Approach

[13] Yu T., Modeling Occupancy Behavior for the Energy Efficiency and Occupants Comfort Management in Intelligent Buildings. Machine Learning and Applications (ICMLA), Ninth International Conference (2010), 726-731