IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 8, 2020, accepted December 21, 2020, date of publication January 1, 2021, date of current version January 21, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3048891

A Unified Methodology to Predict Wi-Fi Network Usage in Smart Buildings

GUILHERME HENRIQUE APOSTOLO[1,2], FLAVIA BERNARDINI[2], LUIZ C. SCHARA MAGALHÃES[1,3], AND DÉBORA CHRISTINA MUCHALUAT-SAADE[1,2]

[1]MídiaCom Research Lab, Niterói 24210-240, Brazil
[2]Institute of Computing, Universidade Federal Fluminense, Niterói 24210-310, Brazil
[3]Telecommunications Engineering Department, Universidade Federal Fluminense, Niterói 24210-240, Brazil

Corresponding author: Guilherme Henrique Apostolo (guilherme_apostolo@midiacom.uff.br)

This work was supported in part by Rio de Janeiro State Research Foundation (FAPERJ), in part by Coordination for the Improvement of Higher Education Personnel (CAPES), in part by CAPES-Print, and in part by National Council for Scientific and Technological Development (CNPq).

ABSTRACT People usually spend several hours per day inside buildings, and they require great amounts of energy and resources to operate. Although there are numerous studies about smart buildings, there is still a need for new intelligent techniques for efficient smart building management. This paper proposes the use of Wi-Fi network association information as a basis for the design of intelligent systems for smart buildings. We propose a unified experimental methodology to evaluate machine learning (ML) models on their capacity to accurately predict Wi-Fi access point demand for energy-efficient smart buildings. The evaluation involves the use of multiple classification and regression models using a variety of configurations and algorithms. We conducted an experimental analysis using our proposed methodology to determine which ML models provide the best performance results using data collected from a large scale Wi-Fi network located at Fluminense Federal University (UFF) over a period of 6 months. The proposed methodology enables the user to evaluate and to create ML models for energy efficient smart building management systems. We achieved 86.69% accuracy for occupancy prediction using classification techniques and RMSPE (Root Mean Squared Percentage Error) of 0.29 for occupancy count prediction using regression techniques.

INDEX TERMS Access point occupancy prediction, energy saving, machine learning, smart buildings, Wi-Fi networks.

NOMENCLATURE

S	Dataset
$\mathcal{D}$	Fixed and unknown distribution
$\mathbf{x}_i$	Feature vector of the ith instance
x_i^M	Value of the Mth feature of the feature vector of the ith instance
$\mathbf{Y}_i$	Set of labels associated with the ith instance
L	Set of possible label values
l_q	qth label in the label set
$\|L\|$	Label cardinality
t	Set of time slots
t_j	jth time slot in the time slot set
t_{max}	Maximum time slot value
$Y_i^{t_j}$	Label value in the jth time slot of the ith instance
S_{t_j}	Training set of the jth time slot
S'_{t_j}	Test set of the jth time slot
A_{t_j}	Accuracy of time slot t_j
$TP_i^{t_j}$	True positive value of the jth time slot of the ith instance
$FP_i^{t_j}$	False positive value of the jth time slot of the ith instance
$TN_i^{t_j}$	True negative value of the jth time slot of the ith instance
$FN_i^{t_j}$	False negative value of the jth time slot of the ith instance
P_{t_j}	Precision of time slot t_j
R_{t_j}	Recall of time slot t_j
$F1_{t_j}$	F1-score of time slot t_j
M	Set of metrics used
$RMSPE_{t_j}$	Root Mean Squared Percentage Error of time slot t_j
$RMSE_{t_j}$	Root Mean Square Error of time slot t_j
$MAPE_{t_j}$	Mean Absolute Percentage Error of time slot t_j
P_{ext_on}	Access point external power when the wireless network interface is switched on
P_{ext_off}	Access point external power when the wireless network interface is switched off

The associate editor coordinating the review of this manuscript and approving it for publication was Antonino Orsino.

t_{on} Time that the access points stayed with their wireless interface switched on

t_{off} Time that the access points stayed with their wireless interface switched off

I. INTRODUCTION

Buildings play an important role in our lives. People usually spend in average 20 hours per day inside buildings [1]. Also, the number of inhabitants in urban areas is quickly increasing [2]. Since buildings are heavily occupied, they require great amounts of energy and resources to operate. As a consequence, there are numerous studies about smart buildings [3]–[5], specially on the creation of low cost, efficient smart building management systems.

The key concept behind smart building management systems is the preemptive control of building infrastructure in order to save resources such as lighting, Heating, Ventilating and Air Conditioning (HVAC), elevators and even network infrastructure [3], [5], [6]. Some building management systems do not require precise occupancy information to be functional and capable of saving energy, especially HVAC systems, by using fixed building control schedules [7]. Several studies have demonstrated that occupancy information could help to reduce energy consumption in buildings, specially in non-residential buildings [3], [8], which operate under more predictable schedules [3].

Wi-Fi networks can be used to conduct occupancy detection or occupancy counting for buildings. The ubiquity of large-scale Wi-Fi networks on non-residential buildings turns them into an excellent source of information with no additional cost [3], [4], [9]. There are several studies that use Wi-Fi infrastructure and machine learning techniques to create prediction models for smart building management. Some of them collect information on the building areas occupancy history to predict if they are occupied or not (occupancy detection) [3], [5], [8], [10]. Others use Wi-Fi information to predict the occupancy count of some building areas [1], [4], [6], [9]. In this scenario, several studies use Wi-Fi infrastructure combined with machine learning methods to predict occupancy of building areas, floors and rooms [1], [3]–[5], [8], [10], [11]. They do not necessarily use the association history information from the Wi-Fi network to build their dataset and create prediction systems, but rather other information such as channel utilization or bandwidth [1], [3]–[5], [8], [10]–[13]. Those studies used single-label or multi-label machine learning classification models and artificial neural networks (ANNs) to address the occupancy detection problem using Wi-Fi association history and developed mechanisms that decide whether an AP should be turned on or off [1], [6], [9], [12], [14]–[16]. There are others that use Wi-Fi association data to create single-label machine learning regression models to estimate the occupancy count that can also be used on Wi-Fi AP energy saving mechanisms [3], [17]. However, those models are mostly used to develop HVAC scheduling systems [5], [10], [11], [18]. On the other hand, none of them has used multi-label regression methods for occupancy count or compared and evaluated single-label and the multi-label methods to classification models to determine which would have greater accuracy on occupancy detection. Thus, we fill this gap with our work.

According to Cui *et al* [19], energy consumption in a Wi-Fi network is considerable. University wireless networks display a bimodal periodic behavior with daily and weekly cycles, and Wi-Fi Access Points (APs) may stay unused for extensive periods of time [6], [20], [21]. These long idle periods represent a considerable energy waste that presents an excellent optimization opportunity. That scenario allows the use of machine learning prediction models capable of delivering occupancy demand predictions for network APs throughout the day [6]. The Wi-Fi network controller can switch off the network interface of unused APs during idle time slots based on those predictions. Some wireless network controllers have limited CPU power making it unfeasible to collect and predict occupancy in real time, therefore requiring these systems to make predictions based exclusively on past information. But even such networks could benefit from intelligent systems and few to no adjustments would be required. Those systems can aid both wireless network energy savings and also other building systems such as elevator scheduling. Therefore our scenario requires an analysis on how machine learning algorithms are capable of looking at the future based on previous information and giving accurate predictions about the Wi-Fi network demand in both occupancy detection and count methods. Such mechanism would use the presence and number of users to create a final demand prediction that could be used to group neighbor APs, and choose which APs should be active to cope with the actual network demand.

The main contribution of this paper is the proposal of a unified experimental methodology based on machine learning to evaluate classification and regression models about their capacity to accurately predict access point demands for energy-efficient smart buildings. Our proposed experimental methodology considers several machine learning algorithms and methods for constructing distinct classification and regression models using multiple input and output configurations. We did an experimental analysis using our unified methodology to determine which models provide the best results or are the most suitable for an energy-efficient wireless network management system.

In order to conduct our experimental analysis, we built a dataset using real user data collected from a subset of the APs of the Fluminense Federal University (UFF) wireless network located in a specific building of the Engineering campus, which has 5 floors of classrooms, over a period of 6 months, from April to September 2018. We used both single-lable machine learning algorithms and multi-label machine learning methods (Binary Relevance (BR) and Classifier/Regressor Chain (CC/RC)). We built multiple models for various output modeling, i.e., individual and collective APs prediction; single and multi-label models for

TABLE 1. Related work comparison.

	Classification		Regression		Models	
Authors	Multi-label	Single-label	Multi-label	Single-label	Collective	Individual
Balaji et al. [5]		X			X	
Zou et al. [3]		X			X	
Zou et al. [1]		X			X	
Fang et al. [6]		X				X
Lyu et al. [12]		X				X
Sangogboye et al. [9]	X					X
Donevski et al. [23]				X	X	
Trivedi et al. [10]				X	X	
Wang et al. [18]				X	X	
Hobson et al. [11]				X		X
Vallero et al. [24]				X	X	X
Our work	X	X	X	X	X	X

predicting occupancy in multiple time slots; and multiple features for input configurations. We evaluated these classification and regression models based on many classification and regression metrics using our dataset. We show how our proposed unified methodology can help to select prediction models using machine learning for occupancy prediction based on performance metrics evaluation for distinct scenarios.

This paper is organized as follows: Section II discusses related work. Section III presents the multi-label classification and regression methods and performance metrics. Section IV presents UFF's Wi-Fi network characteristics, the data collection process and the data analysis. Section V explains our proposed methodology. Section VI presents our experimental analysis. Section VII discusses the results based on distinct usage scenarios. Finally, Section VIII presents conclusions and future work.

II. RELATED WORK

The information collected from Wi-Fi networks, used to build a dataset and create a prediction system, is not always the same, as can be observed in [1], [3]–[5], [8], [10]–[13], [22]–[24]. However, the key concept behind those studies is collecting data about the Wi-Fi network to create a detection or counting system using machine learning algorithms. Those decision support systems provide information for an energy saving management mechanism that controls building infrastructure based on its demand, such as the Wi-Fi network itself or HVAC systems.

Both classifier and regression model are used on Wi-Fi Resource On Demand (ROD) management systems. Those ROD systems are capable of controlling the energy state of Access Points (APs) and turn off the unnecessary APs during day periods based on the predicted occupation [15], [16], [22], [25]. Some studies used classification models to address the Wi-Fi occupancy detection problem and developed ROD mechanisms [3], [6], [12]. Some other studies use single-label machine learning classification methods and ANNs using Wi-Fi data to control building lights [1], [5]. The work presented in [9] used algorithm adaptation multi-label methods to deal with the classification problem for HVAC systems. Regression models using Wi-Fi data to give an estimated users count are mostly used in HVAC scheduling systems [10], [11], [18], but some studies have also used regression models to develop ROD strategy mechanisms [23], [24].

Table 1 compares related work about how they build occupancy prediction models. We can see in the table that most of the occupancy detection studies use single-label classifiers and that none of the occupancy count studies use multi-label regressors, but only single-label ones. Also, those studies did not compare and evaluate single-label and multi-label methods to determine which would give the best predictions, as our work does. It is worth mentioning that while the study of Vallero *et al.* [24] use and compare both individual and collective models, it does not compare them using the same machine learning algorithms, but it rather compares collective and individual models using several machine learning algorithms. Moreover, Table 1 shows that there was no consensus on whether to use collective or individual models to give predictions and that no other study compares them, as our work does.

Finally, there are also some studies where pieces of information related to weather and season of the year were added to the occupancy information, in order to help on decision support systems for smart buildings [7], [9], [18], [26]. None of these studies have developed a methodology where the significance of this information is evaluated though.

Our work presents a unified experimental methodology to evaluate classification models used for occupancy detection and regression models used for occupancy count where several machine learning methods, input configurations, types of model construction and machine learning algorithms are assessed. The main goal of the assessment is to determine which of these parameter combinations is the most suitable and precise to give occupancy predictions. Our methodology evaluates and compares multi-label and single-label methods using several machine learning algorithms, collective and individual model construction schemes and the significance of input parameters. Another major contribution of our experimental methodology and analysis is that it does not require real-time data acquisition for forecasts.

III. MULTI-LABEL AND SINGLE-LABEL LEARNING METHODS

In a general supervised learning scenario, a dataset $S = \{(\mathbf{x}_1, Y_1), \ldots, (\mathbf{x}_N, Y_N)\}$ is given to the learning method, with fixed and unknown distribution $\mathcal{D}$. Each instance $\mathbf{x}_i$ is a vector of the form $\mathbf{x}_i = (x_i^1, \ldots, x_i^M)$. Each value $(x_i^1, \ldots, x_i^M)$ is relative to each feature $(X_1, \ldots, X_M)$. Y is a special feature called class. $Y_i, i = 1, \ldots, N$, represents a set of labels associated to each instance $\mathbf{x}_i$. If all sets $Y_i, i = 1, \ldots, N$, have only one value, the problem is called single-label. So, in single-label problems, machine learning algorithms have only one possible output prediction. However, some machine learning problems cannot be treated as a single-label problem [27]. There are cases, such as movie classification, where a movie can be classified as action and fiction simultaneously [28]. Multi-label machine learning algorithms and methods are those capable of dealing with more than one exclusive output. In other words, if the sets Y_i contain one or more values, the problem is called multi-label. In a multi-label problem, a set $L = \{l_1, \ldots, l_q\}$ is given, such that all $Y_i \in L$.

There are many distinct methods to tackle multi-label problems. Problem transformation is the simplest and the most often used, converting the multi-label problem with L labels into L single-label problems, *i.e.*, each label $l_q \in L$ is turned into a feature, composing a set of features $l_q, q = 1, \ldots, Q$. The cardinality of L is denoted by $|L|$. Thus, each feature l_q is a class associated with the set of instances $\mathbf{x}_i$ to be given to a single-label classification algorithm [29]. In our scenario, for modeling occupancy prediction as a multi-label problem, L represents the time slots for predicting occupancy during a day. For instance, considering a set of time slots $t = \{t_1, .., t_{max}\}$, if each time slot has 10 min, then $t_{max} = 144$ and $|L| = 144$. Therefore, to each instance $\mathbf{x}_i$ and label (or time slot) l_q, we can associate a value $Y_i^{t_j}$ that represents: (i) a value of the set $\{0, 1\}$, indicating absence or presence of people in an AP for time slot t_j, defining a classification problem; or (ii) the number of people associated to an AP for time slot t_j, defining a regression problem.

Classifier or Regressor Chain (CC or RC) methods can be used, as they benefit from label correlations. It is expected that CC or RC achieve more accurate results than Binary Relevance (BR) when there are dependencies among labels [29]. Like BR, CC or RC also build a unique model for each label, but the models are sorted in a chain order. Each model input is composed by the domain features and the labels that precede the label being predicted by the model, forming a chain structure.

Artificial Neural Network (ANN) models have proven to be successful in a number of prediction applications [26]. According to Gardner and Dorling [30], a MultiLayer Perceptron (MLP) is an ANN where the neurons are interconnected and grouped into layers. Neuron connections are weighted and their output signal is an activation function of the sum of its weighted inputs [30]. MLP allows a single ANN to have a single or multiple output targets easily turning the MLP into a multi-label prediction model.

Several metrics can be used for evaluating the classification results. In this work, we use specific label-based micro averaged metrics [28] for both single-label and multi-label models. So, we evaluate occupancy predictions for each time slot and then average those results to get an overall view. Considering a training set $S_{t_j} = \{(\mathbf{x}_1, {Y_1}^{t_j}), \ldots, (\mathbf{x}_N, {Y_N}^{t_j})\}$ collected in an interval of N days; a test set $S'_{t_j} = \{(\mathbf{x}'_1, {Y'_1}^{t_j}), \ldots, (\mathbf{x}'_{N'}, {Y'_{N'}}^{t_j})\}$ collected in an interval of N' days after N days; time slots in a day $t_j \in t$ (if each time slot has 10 min then $t_{max} = 144$); and $\mathbf{h}(\mathbf{x}, t_j)$ a model constructed using S labeled using time stamp t_j, $t_j \in t$, and to be evaluated with S' also labeled using time stamp t_j, $t_j \in t$, we can define time slot accuracy A_{t_j} for each time slot $t_j \in t$ as shown in Eq. 1, which calculates the accuracy of correctly predicting presence or absence detection in each time slot in a day, averaged by the number of N' days.

$$A_{t_j} = \frac{1}{N'} \sum_{i=1}^{N'} \mathbf{h}(\mathbf{x}, t_j) = {Y'_i}^{t_j}, t_j \in t \tag{1}$$

Considering the true positive value ${TP_i}^{t_j}$ of an instance i for a time slot t_j as 1 if $\mathbf{h}(\mathbf{x}, t_j) = {Y'_i}^{t_j}$ and $\mathbf{h}(\mathbf{x}, t_j) = 1$, or 0 otherwise; the false positive value ${FP_i}^{t_j}$ of an instance i for a time slot t_j as 1 if $\mathbf{h}(\mathbf{x}, t_j) \neq {Y'_i}^{t_j}$ and $\mathbf{h}(\mathbf{x}, t_j) = 1$, or 0 otherwise; true negative value ${TN_i}^{t_j}$ of a instance i for a time slot t_j as 1 if $\mathbf{h}(\mathbf{x}, t_j) = {Y'_i}^{t_j}$ and $\mathbf{h}(\mathbf{x}, t_j) = 0$, or 0 otherwise; and the false negative value ${FN_i}^{t_j}$ of a instance i for a time slot t_j as 1 if $\mathbf{h}(\mathbf{x}, t_j) \neq {Y'_i}^{t_j}$ and $\mathbf{h}(\mathbf{x}, t_j) = 0$, or 0 otherwise, we can define Precision P_{t_j}, Recall R_{t_j} and F1-score $F1_{t_j}$ metrics. Those metrics are calculated for each time slot t_j and defined respectively by Eqs. 2, 3 and 4.

$$P_{t_j} = \frac{\sum_{i=1}^{N'} TP_i^{t_j}}{\sum_{i=1}^{N'} TP_i^{t_j} + FP_i^{t_j}}, t_j \in t \tag{2}$$

$$R_{t_j} = \frac{\sum_{i=1}^{N'} TP_i^{t_j}}{\sum_{i=1}^{N'} TP_i^{t_j} + FN_i^{t_j}}, t_j \in t \tag{3}$$

$$F1_{t_j} = \frac{2 \times P_{t_j} \times R_{t_j}}{P_{t_j} + R_{t_j}}, t_j \in t \tag{4}$$

We also calculate an overall metric for each of these metrics (Eq. 5), which is the mean of the corresponding metric considering all the set t of time slots. This allows an overview of **h** prediction performance for the classification problem. Thus, M in Eq. 5 can be either A, P, R or $F1$ metric.

$$\overline{M} = \frac{1}{t_{max}} \sum_{j=1}^{t_{max}} M_{t_j} \tag{5}$$

Several metrics can be used for evaluating regressors. Consider the same definitions described before, except that $Y_{i'}^{\prime t_j}, i' = 1, \ldots, N'$ now represents the number of people associated to an AP in a time slot t_j. So, we can use $RMSE_{t_j}$ (Root Mean Square Error), $RMSPE_{t_j}$ (Root Mean Squared Percentage Error) and $MAPE_{t_j}$ (Mean Absolute Percentage Error) metrics, defined respectively by Eqs. 6, 7 and 8, calculated for each time slot t_j, where $\overline{Y'}_{t_j} = \frac{1}{N'} \sum_{i'=1}^{N'} Y_{i'}^{\prime t_j}$.

$$RMSE_{t_j} = \sum_{i=1}^{N'} (Y_i^{\prime t_j} - h(x_i', t_j))^2 \tag{6}$$

$$RMSPE_{t_j} = \frac{\sum_{i=1}^{N'} (Y_i^{\prime t_j} - h(x_i', t_j))^2}{\sum_{i=1}^{N'} (Y_i^{\prime t_j} - \overline{Y'_{t_j}})^2} \tag{7}$$

$$MAPE_{t_j} = \frac{\sum_{i=1}^{N'} |Y_i^{\prime t_j} - h(x_i', t_j)|}{\sum_{i=1}^{N'} |Y_i^{\prime t_j} - \overline{Y'_{t_j}}|^2} \tag{8}$$

$\overline{RMSE}$ (Eq. 9) is an overall metric, calculated by the mean of $RMSE_{t_j}$ using the entire set t. The overall metric for $MAPE$ or $RMSPE$ can also be calculated by Eq. 5, where M can be $\overline{MAPE}$ or $\overline{RMSPE}$.

$$\overline{RMSE} = \sum_{i=1}^{N'} \sum_{j=1}^{t_{max}} (Y_i^{\prime t_j} - h(x_i', t_j))^2 \tag{9}$$

IV. UFF's WI-FI NETWORK DATA COLLECTION AND ANALYSIS

UFF's Wi-Fi network is based on the SCIFI system [31]. The SCIFI system is composed of a smart management and monitoring central controller unit, called SCIFI controller, and low-cost off-the-shelf APs, operating under a custom made open source OpenWRT firmware [32]. The SCIFI controller coordinates data acquisition from system logs, and sets channel and transmission power for each AP in order to minimize interference. SCIFI is used at UFF, UFOP (Ouro Preto Federal University) and Brazilian Navy laboratories.

In this work, we used 28 APs spread over 5 floors of the H building at UFF's Engineering Campus. We chose the H building because it is fully composed by classrooms and follows a strict occupation schedule. We collected data from 6 months, between April and September 2018.

Each AP sends management and control events to the SCIFI controller. Thus, we filtered log files to collect association and disassociation or deauthentication events information for the target APs. Association events mark the beginning of an active connection between the AP and a user station, while disassociation events mark its end. We observed that disassociation events did not always appear in the log data, however we also observed that whenever disassociation and deauthentication of mobile stations message appeared in the event logs, both occurred approximately in less than 1 second difference between them. Hence, we used deauthentication messages as the end of a connection between a mobile station and an AP, when there was no registered disassociation messages.

A. OCCUPANCY ANALYSIS

Figures 1 and 2 show the average SCIFI network behavior in the H building from April to September 2018. It is possible to observe the daily and weekly average occupancy. Figure 1 shows that APs are mostly idle between 0 and 6AM. It also shows a slowly increasing occupation for time slots between 6 and 9AM. That slow build can be explained by the lecture time schedules for the H building, which start at 7AM, but most of them start at 9AM, and the last lectures end at 10PM. Morning classes start at odd hours, and afternoon classes at even times, with an hour interval between 1 and 2PM. Figure 1 shows that AP's occupation during university weekdays is higher than the occupation at holidays and weekends. The occupation for holidays are slightly higher than those for weekends. These results were unforeseen, but can be explained by the H building usage during student vacations for summer/winter courses and special activities.

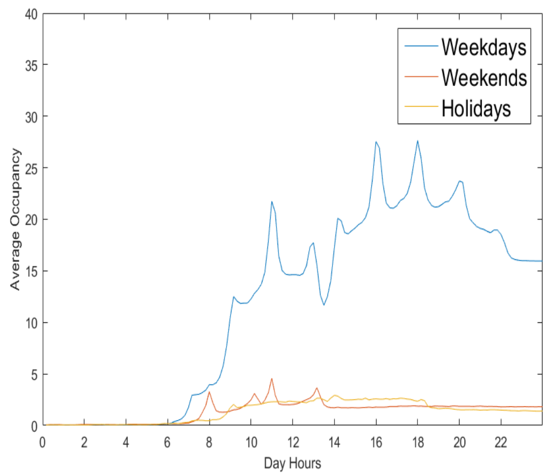

FIGURE 1. Average day occupancy comparing working days, weekends and holidays.

Figure 2 shows that the AP demand is higher during weekdays than during weekends. The average occupancy reaches its highest on Tuesdays, Wednesdays and Thursdays. Whilst smaller than the other weekdays, Saturday's average demand is relatively high when compared to Sunday. One explanation can be that the building is used on some Saturdays for exams and other special activities. For a classroom building such as the H building, these results were expected. Even though most APs in the H building remain unoccupied for long periods of time, we noticed that some APs remain with a residual number of devices connected to it during closing hours. One possible explanation is that the H building still has appliances, such as computers, and university staff members, such as the

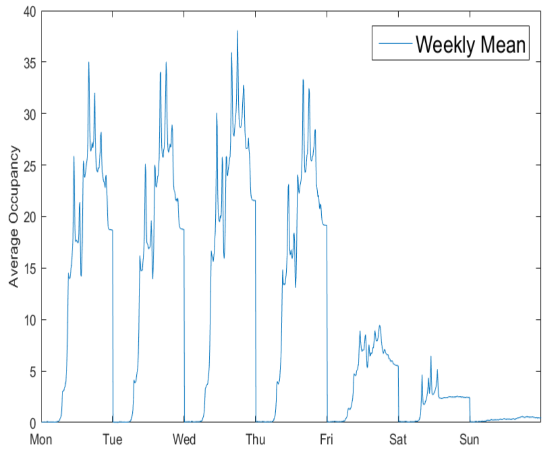

FIGURE 2. Average week occupancy.

campus security, that are still present in the building during off hours and days. That explains why we can not simply assume that all APs are unused during those hours.

V. PROPOSED METHODOLOGY

Figure 3 shows a schema of our proposed unified methodology and its major steps, which are i) data acquisition and dataset construction; ii) input configuration; iii) regression and classification model configuration; iv) model selection.

The first step, shown in the upper part of the figure, is to prepare four datasets to be used for the evaluation of classification and regression prediction models. Then, in the second step, we use several input feature configurations, training set constructions and distinct single-label and multi-label machine learning methods to build our classifiers and regressors, in order to evaluate the significance of these characteristics for prediction models.

In the third step, we build single and multi-label classifiers capable of predicting the occupancy states for network APs and/or the construction of single and multi-label regressors capable of predicting the occupancy count for network APs. For multi-label classification, we propose using BR and CC problem transformation methods and Multilayer Perceptron ANN to produce forecasts. For multi-label regression, we propose using BR and RC problem transformation methods and Multilayer Perceptron ANN for multi-label to produce those predictions.

Finally, in the last step shown in Figure 3, an evaluation using multi-label and single-label metrics helps the selection of a model that provides the best performance results and that can be used in smart building energy-efficient systems for several purposes.

Our methodology is validated throughout experimental tests with UFF's SCIFI network data. Our tests demonstrate how it helps deciding the most suitable method to be used for energy-efficient smart buildings. In our experiments, dataset transformations, classification and regression model construction and measurements were developed using Python scikit-learn API [33] and Pandas [34].

A. DATASET CONSTRUCTION

We have filtered and processed event logs to select information about the connection status between mobile stations and APs. Our datasets follow the work of Sangogboye, Imamovic and Kjærgaard [9] and Balaji *et al* [5]. We divided a day into 144 (10 minutes) time slots, and computed the number of devices associated to an AP in each time slot by increasing the number for each station association event and decreasing it for each disassociation event. The datasets[1] show occupancy count and detection for each AP over a period of 6 months, from April to September 2018.

In the single-label datasets, each instance has only one output feature representing a specific date and time interval occupation count. The single-label dataset contains the following input features: Month, Day, Day of the Week, Holiday, APid, Hour, Minute. The multi-label datasets have each instance representing one specific date and 144 output features representing the time intervals of a day occupation count. The multi-label dataset contains the following input features: Month, Day, Day of the Week, Holiday, APid.

Month and Day are numeric and show the instance date. Day of the week is categorical and indicates one of the 7 week days. Holiday is boolean and indicates if the day is a normal semester day with lectures (False) or a public holiday or university vacation day (True). AP Identification (APid) carries the access point identification number and it informs to which specific AP the occupancy history belongs. Hour and Minutes are also numerical and are only present in the single-label datasets. The Hour input feature ranges from 0 to 23 representing day hours. The Minute feature ranges from 0 to 50 in 10 minute steps. Although we could have combined Hour and Minute features to create a time interval feature ranging from 0 to 144, we decided to keep semantic information given by the hour/minute tuple.

On occupancy detection datasets, we are only interested in binary classification (whether the AP has someone associated or not), so we applied a label binarization filter to our dataset outputs, in order to transform each numeric occupation count into a boolean output feature. To be classified as occupied (value 1) for a 10 minute time interval, the AP needs to have at least one mobile station associated to it. If no mobile station tries to associate to that AP during the whole duration of that time slot, the AP is considered unoccupied (value 0). The single and multi-label occupancy detection datasets have the same input features.

B. SINGLE-LABEL AND MULTI-LABEL CLASSIFICATION ANALYSIS

We evaluated multiple types of classification model constructions, with varying training and testing sets. We trained collective models where only one classifier was trained with information regarding all APs and responsible for predicting the occupancy detection of all APs. We also trained individual

[1] The datasets are available at https://github.com/midiacom/UFF-SCIFI-Datasets

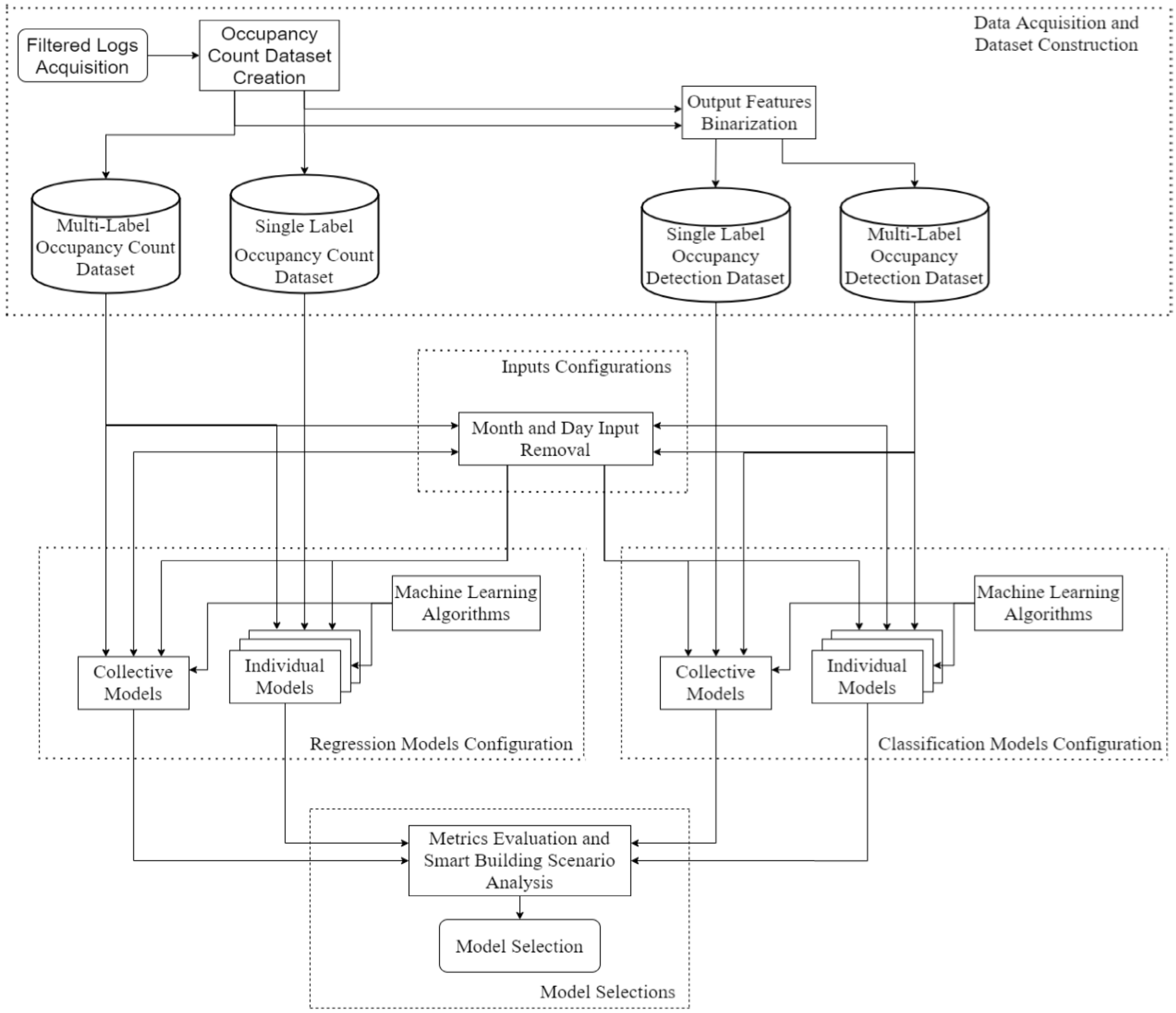

FIGURE 3. Our proposed methodology.

classification models where multiple classifiers were trained only using information regarding one specific AP and responsible for that AP occupancy detection prediction. We built collective MLP ANN multi-label (ML) and single-label (SL) classifiers for our tests. Our goal with these distinct single and multi-label model construction was to evaluate if the occupancy detection of one AP could benefit from information from other APs, to determine if an AP individual information is capable of giving satisfactory detection predictions and which method has the best performance among those tested.

These collective and individual multi and single-label classifiers were also tested using multiple input feature configurations. We decided to evaluate if Month and Day features were significant to our model predictions. Month and Day features give date information to the classification models, which could benefit their predictions giving seasonal insights. On the other hand, more features can also represent more noise and increase the size of the classification data, which can consequently turn into waste of space and insignificant accuracy enhancement. Therefore, all classifiers were trained with and without Month and Day features.

Our label features are used respecting the time interval order for constructing the chain in the CC method. Therefore, our feature chain goes in crescent order from T_0 to T_{143}. Our time sequenced output features helped chain selection order in CC, because finding label order can be challenging [29]. We used Decision Tree (DT), K-NN and Random Forest (RF) machine learning algorithms for our SL classification models, as they present the best single-label Wi-Fi occupancy detection results according to Fang et al [6]. Sangogboye, Imamovic and Kjærgaard [9] also stated that these algorithms were among the best algorithms in their ML method. We used default parameters values for DT and RF and we used $K = 5$ for K-NN.

We also built ANN MLPs. Table 2 shows the MLP hyper parameters selected for both SL and ML classification

TABLE 2. MLP ANN parameter values.

MLP Parameter	Best SL Parameter	Best ML Parameter
Hidden layer size	400	900
Alpha	0.0001	0.001
Learning rate	invscaling	invscaling
Activation	logistic	relu
Max iteration	1000	1000
Random state	1	1

models after a search over a list of possible values for hyper parameters. We used the grid search algorithm GridSearchCV present in scikit-learn API [33]. Other non-listed parameters kept their default values.

To evaluate the performance of these models, we apply a train/test split on our datasets. The order of the collected data must be respected both for training and testing. So, dataset instances from April to August were used for training, and September dataset instances were used for testing the models. We used 4 metrics to evaluate our classification models: A_{t_j}, P_{t_j}, R_{t_j} and $F1_{t_j}$, as well as their overall versions.

C. SINGLE-LABEL AND MULTI-LABEL REGRESSION ANALYSIS

For occupancy count, we also tested multiple types of regression model construction, with various training and testing sets. We trained collective and individual regressors using distinct training sets. These collective and individual multi and single-label regressors also were tested having several input feature configurations. Consequently those multi-label (ML) and single-label (SL) collective and individual regressors were trained with and without the Month and Day features. Those regression model constructions evaluate if the occupancy count system could benefit from information from other APs, determine if an AP individual information is capable of giving satisfactory results and evaluate if Month and Day features were significant for predictions.

The output label chain in RC methods is the same used in CC. We used DT, K-NN, RF and the XG optimized gradient boosting SL learning regression algorithms. Later on, we decided to construct collective MLP ANN, support vector machine (SVM) and stochastic gradient descent (SGD) SL and ML regressors. But since the occupancy count data presents a high variance, these regressors had their input and output data normalized. We also decided to test the K-NN algorithm with normalized input and output data. The MLP hyper parameters selected after an extensive search for both SL and ML regression models are the ones shown in Table 2.

Analogously to the classifier evaluation, we also applied a train/test split on our datasets. Dataset instances from April to August were used for training, and September dataset instances were used for testing the models. We used three metrics to evaluate our regression models: $RMSE_{t_j}$, $RMSPE_{t_j}$ and $MAPE_{t_j}$, as well as their overall versions.

VI. EXPERIMENTAL ANALYSIS

This section shows the results of our experimental analysis. We analyze which machine learning method, algorithm, model construction type and input combinations are more suitable to scenarios where Wi-Fi data can be used for smart building systems.

A. CLASSIFIER ANALYSIS

In what follows, we show the experimental analysis for the occupancy detection problem. The models were constructed using a combination of four distinct parameters: the SL method and 2 distinct ML (BR and CC) machine learning methods; 2 distinct types of model construction, which can be collective (Col) or individual (Ind); 2 distinct input configurations, one composed by APid, holiday and weekday features (APHDWD) and other by all features (ALL), including AP Id, holiday, weekday, day and month features; and 3 distinct machine learning algorithms (RF, DT and K-NN) for constructing both SL models and the base classifiers of the ML methods. We also constructed 2 collective SL and 2 collective ML MLP ANNs, one using APHDWD features and other using ALL features. These combinations result in 40 distinct models. In order to guide our analysis, we firstly compare BR and CC ML methods. Then, we compare the best ML method against the SL method. We then evaluate types of model construction, algorithms and inputs. Finally, we evaluate if there is any observable advantage of one combination of parameters over the others.

1) MULTI-LABEL METHODS

We selected the best results from the 40 evaluated models. Figure 4 depicts the accuracy A_{tj} of the best machine learning algorithm for each possible BR and CC ML classification model parameter combinations. We can see that BR models have better accuracy results than CC, as well as they drastically decrease from 6 to 8AM for both methods.

CC performance can be explained by the unpredictable AP occupancy from 6 to 8AM as seen in Figure 1. As the occupancy and idleness occurrence in those time slots are very alike and the states occur almost randomly, it is harder for classifiers to give a correct occupancy prediction for them, which leads to worse accuracy. That accuracy loss introduces a greater error on the label feature prediction and consequently affects the rest of the chain since the next time slots take the previous results into consideration. Because BR does not take the previous prediction into account, those prediction errors do not propagate.

Table 3 shows the overall metrics $\overline{A}$, $\overline{P}$, $\overline{R}$ and $\overline{F1}$ for the best assessed models. From Table 3, it is clear that the BR method got better overall results than the CC method. Metric evaluation also shows that models using only APHDWD as input features present better results than using ALL features. Thus, this result indicates that, for our data, seasonal information is not a significant feature for ML classification models. Metric evaluation also shows that there is no significant

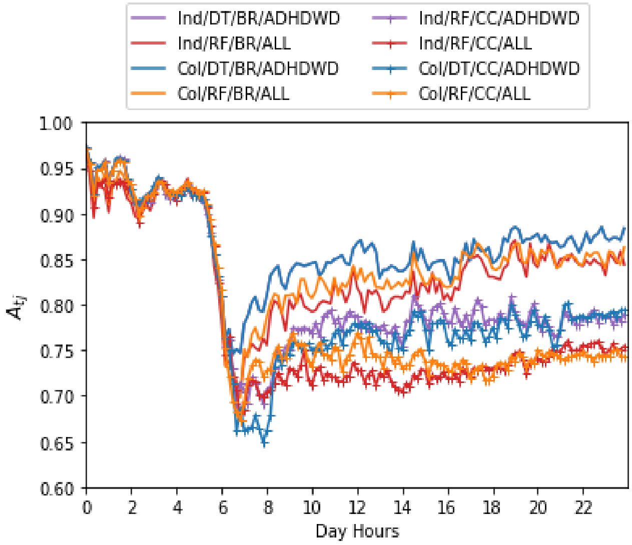

FIGURE 4. Accuracy A_{tj} for several BR and CC ML methods and parameter configurations.

TABLE 3. Classification performance results for BR and CC ML methods.

Constructed Models	$\overline{A}$	$\overline{P}$	$\overline{R}$	$\overline{F1}$
Col/DT/BR/APHDWD	**0.8669**	**0.8662**	**0.8960**	**0.8808**
Col/DT/CC/APHDWD	0.8025	0.8683	0.7548	0.8076
Col/RF/BR/APHDWD	0.8631	0.8570	0.9010	0.8784
Col/RF/CC/APHDWD	0.8201	0.8536	0.8115	0.8320
Col/DT/BR/ALL	0.8268	0.8261	0.8671	0.8461
Col/DT/CC/ALL	0.7664	0.7693	0.8207	0.7942
Col/RF/BR/ALL	0.8495	0.8508	0.8804	0.8653
Col/RF/CC/ALL	0.7863	0.9161	0.6724	0.7756
Ind/DT/BR/APHDWD	**0.8669**	**0.8662**	**0.8960**	**0.8808**
Ind/DT/CC/APHDWD	0.8025	0.8683	0.7548	0.8076
Ind/RF/BR/APHDWD	0.8631	0.8566	0.9015	0.8785
Ind/RF/CC/APHDWD	0.8113	0.8469	0.8013	0.8235
Ind/DT/BR/ALL	0.8155	0.8155	0.8581	0.8363
Ind/DT/CC/ALL	0.7782	0.7797	0.8309	0.8045
Ind/RF/BR/ALL	0.8412	0.8412	0.8762	0.8583
Ind/RF/CC/ALL	0.7770	0.8755	0.6924	0.7733

difference between the types of model constructions (Col vs Ind), which indicates that both collective and individual models are equally valid model construction types for occupancy detection.

2) MULTI-LABEL AND SINGLE-LABEL EVALUATION

From the multiple combinations of parameters for constructing the SL and ML models, we chose at least one of the best results of 8 combinations for a deeper analysis. Figure 5 shows the A_{tj} accuracy of these 8 models, where we can notice that there is no significant difference between the ML and SL correspondent models. For instance, the A_{t_j} curve of models Col/DT/BR/ADHDWD, Col/DT/SL/ADHDWD,

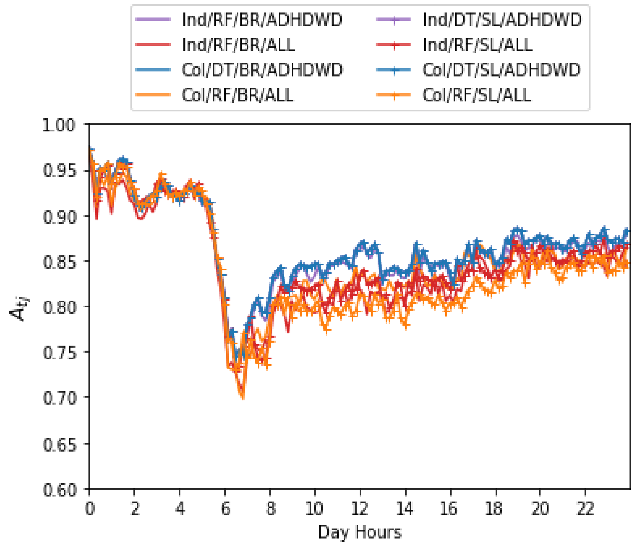

FIGURE 5. Accuracy A_{tj} of ML and SL methods for several parameter configurations.

Ind/RF/BR/ADHDWD and Ind/DT/SL/ADHDWD are quite similar. Also, we could observe that models using only APHDWD features had better results than models using all features (ALL).

Table 4 shows the overall metrics $\overline{A}$, $\overline{P}$, $\overline{R}$ and $\overline{F1}$ for the best ML and SL models. It also shows the results for MLP ANN models. Table 4 demonstrates that the seasonal information do not improve the model predictions. Models using only the APHDWD features had better overall results, which suggest that day and month features carry no significant information about our occupancy data. Our results and the results in [10] comprise the same seasons and yet they showed distinct conclusions about seasonal information. Results reported in [10] showed that seasonal information carries relevant information about the occupancy data. One explanation for that difference can be the low influence of tropical climate at latitude -22.9, where UFF is located.

Table 4 shows that there is no significant difference between ML and SL methods. From Table 4, we can also notice that there is no significant difference between collective and individual models. These conclusions make both machine learning methods and both model construction types equally valid. It is also possible to observe from Table 4 that DT and RF algorithms were the most suited for the occupancy detection problem. Finally Table 4 shows that the ML MLP ANN fails to have comparable results, however the Col/MLP/SL/APHDWD ANN got comparable results to the Col/DT/SL/APHDWD model.

We found that DT and RF machine learning algorithms were the most suited for occupancy detection. Since there was no noticeable difference on the evaluation metrics for ML and SL individual and collective models using the RF and DT algorithms, we decided to evaluate their model sizes in order to compare them. Smaller models are not only simpler to understand, but they also require less memory

TABLE 4. Classification performance results for BR ML and SL methods.

Constructed Models	$\overline{A}$	$\overline{P}$	$\overline{R}$	$\overline{F1}$
Col/DT/BR/APHDWD	**0.8669**	**0.8662**	**0.8960**	**0.8808**
Col/DT/SL/APHDWD	**0.8669**	**0.8662**	**0.8960**	**0.8808**
Col/RF/BR/APHDWD	0.8631	0.8570	0.9010	0.8784
Col/RF/SL/APHDWD	0.8634	0.8567	0.9021	0.8788
Col/MLP/ML/APHDWD	0.8201	0.8536	0.8115	0.8320
Col/MLP/SL/APHDWD	**0.8669**	**0.8662**	**0.8960**	**0.8808**
Col/DT/BR/ALL	0.8268	0.8261	0.8671	0.8461
Col/DT/SL/ALL	0.8277	0.8878	0.8878	0.8498
Col/RF/BR/ALL	0.8495	0.8508	0.8804	0.8653
Col/RF/SL/ALL	0.8388	0.8241	0.8981	0.8595
Col/MLP/ML/ALL	0.7737	0.7359	0.9170	0.8165
Col/MLP/SL/ALL	0.8510	0.8880	0.8339	0.8601
Ind/DT/BR/APHDWD	**0.8669**	**0.8662**	**0.8960**	**0.8808**
Ind/DT/SL/APHDWD	**0.8669**	**0.8662**	**0.8960**	**0.8808**
Ind/RF/BR/APHDWD	0.8631	0.8566	0.9015	0.8785
Ind/RF/SL/APHDWD	0.8633	0.8572	0.9011	0.8786
Ind/DT/BR/ALL	0.8155	0.8155	0.8581	0.8363
Ind/DT/SL/ALL	0.8268	0.8214	0.8747	0.8472
Ind/RF/BR/ALL	0.8412	0.8412	0.8762	0.8583
Ind/RF/SL/ALL	0.8511	0.8445	0.8934	0.8683

space to be stored and are also faster to traverse, which leads to a faster result and smaller CPU requirements to run them. Table 5 shows the mean number of leaves (Numb. of Leaves), depth and their respective standard deviation (Std. Dev.) for all model possible combinations using only APHDWD input features. In this table, we can observe that SL models have a smaller size when compared to ML models. This was expected because the ML BR method consists of a group of individual SL models, each for one specific label. The second conclusion is that DT algorithms are significantly smaller when compared to RF algorithms. This result was also expected since random forests are a collection of decision trees. Finally, we can notice that collective models are larger than individual models. Since individual models train over a smaller part of the dataset they also present smaller sizes. SL and DT algorithms form the best combination to be used in scenarios using our data, because they are simpler and smaller. However, the same cannot be said about individual models over collective models. Individual models are smaller but they only give information about one AP. Depending on the scenario characteristics, the collective model can actually be a better option, such as in our motivation scenario where a central unit is responsible for the management of the whole AP network.

TABLE 5. DT and RF classifier's mean number of leaves and depth size evaluation.

Constructed Models	Mean Numb. of Leaves	Numb. of Leaves Std. Dev.	Mean Depth	Depth Std. Dev.
Col/DT/BR/APHDWD	43409	-	2393	-
Col/DT/SL/APHDWD	**37756**	**-**	**33**	**-**
Col/RF/BR/APHDWD	1918721	-	117475	-
Col/RF/SL/APHDWD	1818221	-	1604	-
Ind/DT/BR/APHDWD	1587	141	601	63
Ind/DT/SL/APHDWD	**1346**	**172**	**20**	**2**
Ind/RF/BR/APHDWD	71903	6690	30805	3490
Ind/RF/SL/APHDWD	62345	8219	932	22

B. REGRESSION ANALYSIS

This section shows the experimental analysis for the occupancy count problem. We evaluated several regression models using SL and ML machine learning methods. 48 models were built using a combination of four distinct parameters: the SL method and 2 distinct ML (BR and RC) methods; 2 distinct types of model construction, which can be collective (Col) or individual (Ind); 2 distinct input configurations, one composed by APHDWD features and other by all features (ALL); and 4 distinct machine learning algorithms (RF, DT, K-NN, XG) for constructing both SL models and the base classifiers of the ML methods. We also constructed 2 collective SL MLP ANNs and 2 collective ML MLP ANNs, using APHDWD features and using ALL features. Additionally, we constructed 12 more collective regression models using a combination of three distinct parameters: 3 distinct machine learning algorithms (SVM, SGD, K-NN); 2 distinct normalized input configurations, one composed by APHDWD normalized features and other by all normalized features (ALL); and 2 machine learning methods (SL and BR). These combinations result in 64 distinct models.

We firstly evaluate BR and RC ML methods. Then, we compare the best ML method against SL methods. We evaluate which model construction type, algorithms and inputs give the best results. Lastly, we evaluate if there is any observable advantage of one method over the others.

1) MULTI-LABEL METHODS

As we tested 64 distinct models, the results shown here are the compilation of the best results found. Figure 6 shows the $RMSE_{tj}$ of the best machine learning algorithm for each possible BR and RC ML regression model parameter combinations. Figure 6 shows that the BR method models have lower $RMSE_{tj}$ values than the RC models and that the $RMSE_{tj}$ results start to significantly increase after 6AM for both methods.

Another interesting observation when comparing Figures 6 and 1 is that $RMSE_{tj}$ increasing behavior is very similar to the occupancy behavior. This means that heavily occupied hours have higher $RMSE_{tj}$ errors. Therefore, $RMSE_{tj}$ is a numerical error metric that alone cannot be enough to evaluate how good the occupancy count predictions are for each time slot individually. Figure 7 shows $RMSPE_{tj}$. We can observe that the BR method got better results than the RC method. BR better performance over RC can be explained by the same reasons we have discussed in Section VI-A1.

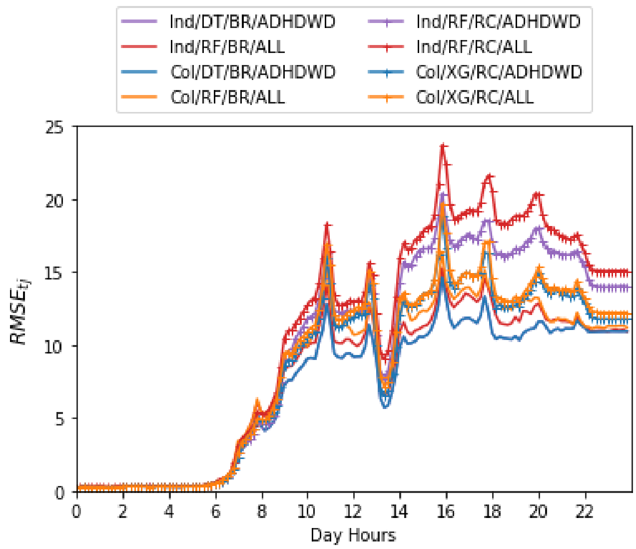

FIGURE 6. ***$RMSE_{tj}$*** **for several BR and CC ML methods and parameter configurations.**

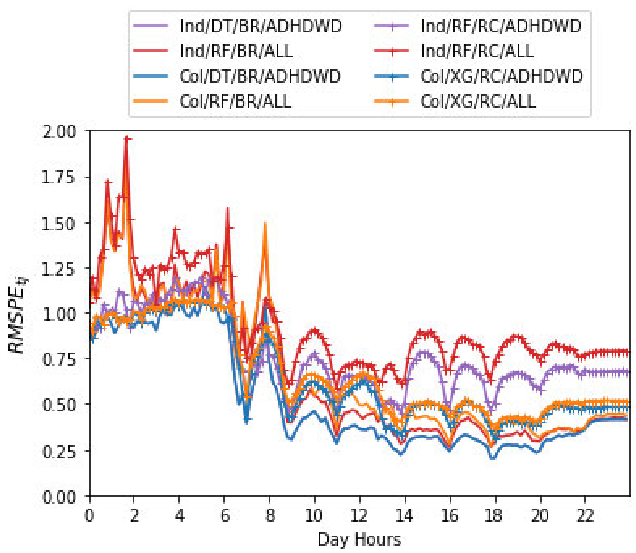

FIGURE 7. ***$RMSPE_{tj}$*** **for several BR and CC ML methods and parameter configurations.**

Comparing Figures 6 and 7, we can also notice that, even though the $RMSE_{tj}$ values are higher for predictions after 9AM, their $RMSPE_{tj}$ values are smaller. Even though the absolute occupancy count error of these time intervals are higher, they are comparatively smaller than the data variance and therefore we can conclude that model predictions are acceptable. The $RMSPE_{tj}$ values presented before 9AM are relatively higher, being almost equal or superior to the variance itself. This happens because these hours real occupancy is low and presents a small variance. Therefore for late-night and early-morning hours, $RMSE_{tj}$ values are comparatively higher than the data variance. However since these hours correspond mostly to closing hours, we can not say that an occupancy count model would not be applicable. Even if we might being doubling the occupancy count values due to prediction errors, the total occupancy count would still be low. So, depending on the scenario and systems, these errors can be easily overcome.

Table 6 shows the overall metrics $\overline{RMSE}$, $\overline{RMSPE}$ and $\overline{MAPE}$ for the best models. Metric evaluation for the regression problem shows that models using only APHDWD input features had better results than the models that used ALL features, which indicates that seasonal information is also not a significant feature for ML regression models. Metric evaluation also shows that there is no significant difference between the model construction types, indicating that both models are equally valid for occupancy count prediction.

TABLE 6. Regression performance results for BR and RC ML methods.

Constructed Models	$\overline{RMSE}$	$\overline{RMSPE}$	$\overline{MAPE}$
Col/DT/BR/APHDWD	**8.4161**	**0.2977**	**0.4189**
Col/DT/RC/APHDWD	13.4306	0.7843	0.6723
Col/RF/BR/APHDWD	8.4223	0.2980	0.4191
Col/RF/RC/APHDWD	11.4110	0.5690	0.5676
Col/DT/BR/ALL	11.2734	0.5379	0.5532
Col/DT/RC/ALL	16.2836	1.1478	0.8887
Col/RF/BR/ALL	9.6314	0.3880	0.4821
Col/RF/RC/ALL	11.6615	0.5896	0.5989
Ind/DT/BR/APHDWD	**8.4161**	**0.2977**	**0.4189**
Ind/DT/RC/APHDWD	12.8850	0.7168	0.6375
Ind/XG/BR/APHDWD	8.4174	0.2978	0.4189
Ind/XG/RC/APHDWD	11.9943	0.6255	0.5803
Ind/DT/BR/ALL	11.0656	0.5151	0.5522
Ind/DT/RC/ALL	15.3127	1.0207	0.8293
Ind/XG/BR/ALL	10.3054	0.4528	0.5841
Ind/XG/RC/ALL	13.2117	0.7667	0.7024

2) MULTI-LABEL AND SINGLE-LABEL EVALUATION

Figure 8 compares $RMSPE_{tj}$ among the best machine learning algorithms for ML and SL regression model construction combinations. It shows that there is no significant difference between ML and SL correspondent models. However it is possible to notice that models using only the APHDWD features had better results than the models that used ALL features.

Table 7 shows the overall metrics $\overline{RMSE}$, $\overline{RMSPE}$ and $\overline{MAPE}$ for the best assessed models. It also shows the results for the MLP ANN models. Table 7 shows that regression models using only the APHDWD features had better overall results, which suggest that day and month features carry no significant information about our occupancy data for the regression problem too. Table 7 overall metric evaluation shows that there is no significant difference between ML and SL methods and that there is no significant difference between collective and individual models, which make both machine learning methods and both model construction

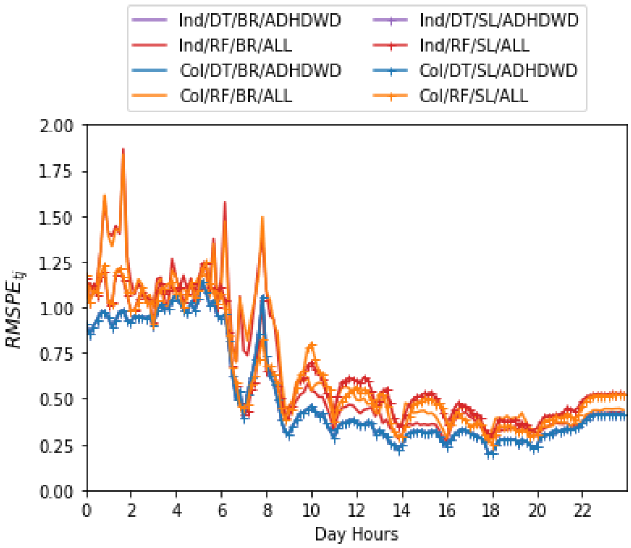

FIGURE 8. $RMSPE_{tj}$ **of ML and SL methods for several parameter configurations.**

TABLE 7. **Regression performance results for BR ML and SL methods.**

Constructed Models	$\overline{RMSE}$	$\overline{RMSPE}$	$\overline{MAPE}$
Col/DT/BR/APHDWD	**8.4161**	**0.2977**	**0.4189**
Col/DT/SL/APHDWD	**8.4161**	**0.2977**	**0.4189**
Col/RF/BR/APHDWD	8.4223	0.2980	0.4191
Col/RF/SL/APHDWD	8.4221	0.2981	0.4192
Col/MLP/ML/APHDWD	10.6320	0.4875	0.6037
Col/MLP/SL/APHDWD	11.2536	0.5489	0.6403
Col/DT/BR/ALL	11.2734	0.5379	0.5532
Col/DT/SL/ALL	10.2350	0.4450	0.4994
Col/RF/BR/ALL	9.6314	0.3880	0.4821
Col/RF/SL/ALL	9.7373	0.3994	0.4718
Col/MLP/ML/ALL	14.3063	0.8955	0.8926
Col/MLP/SL/ALL	14.0249	0.8791	0.8336
Ind/DT/BR/APHDWD	**8.4161**	**0.2977**	**0.4189**
Ind/DT/SL/APHDWD	**8.4161**	**0.2977**	**0.4189**
Ind/XG/BR/APHDWD	8.4174	0.2978	0.4189
Ind/XG/SL/APHDWD	9.0894	0.3625	0.4732
Ind/DT/BR/ALL	11.0656	0.5151	0.5522
Ind/DT/SL/ALL	10.7165	0.4900	0.5231
Ind/XG/BR/ALL	10.3054	0.4528	0.5841
Ind/XG/SL/ALL	10.2894	0.4571	0.5777

equally possible considering performance. It is also possible to observe in Table 7 that DT and RF algorithms were the best machine learning algorithms for occupancy count prediction. Table 7 shows that MLP ANN fails to have comparable results. However it is worth mentioning that the Ind/XG/BR/APHDWD model got comparable results to the DT collective SL model using APHDWD features.

DT and RF machine learning algorithms had better results for occupancy count than the others. Since there was no noticeable difference on the evaluation metrics for ML and SL individual and collective models using these algorithms we also decided to evaluate their model sizes. The model size impacts on memory space and CPU requirements. Table 8 shows the mean number of leaves (Numb. of Leaves), depth and the standard deviation (Std. Dev.) for all model combinations using APHDWD features. This table shows that SL models are smaller when compared to ML models and that the DT algorithm is significantly smaller when compared to the RF algorithm. We can also notice that collective models are bigger than individual models. The reason why these results are expected are the same ones we have discussed in Section VI-A2. SL method and DT algorithm are a better combination to be used in our scenario once they are simpler and smaller than ML methods and the RF algorithm. However, the same cannot be said about individual models over collective models. As we have discussed in Section VI-A2, the collective model can actually be a better option depending on the scenario characteristics.

TABLE 8. **DT and RF regressor's mean number of leaves and depth size evaluation.**

Constructed Models	Mean Numb. of Leaves	Numb. of Leaves Std. Dev.	Mean Depth	Depth Std. Dev.
Col/DT/BR/APHDWD	**48066**	-	-	**2355**
Col/DT/SL/APHDWD	46136	-	-	29
Col/RF/BR/APHDWD	2300688	-	-	115052
Col/RF/SL/APHDWD	2219991	-	-	1465
Ind/DT/BR/APHDWD	**1735**	**170**	**633**	**69**
Ind/DT/SL/APHDWD	1647	205	18	1
Ind/RF/BR/APHDWD	83215	8995	30949	3556
Ind/RF/SL/APHDWD	887	10593	887	17

VII. FURTHER DISCUSSION ON OUR METHODOLOGY AND RESULTS

While other authors have analyzed how multiple machine learning algorithms may change the model prediction results, all studies we have seen in literature did that using only a specific ML or SL method with a specific model construction type and input configuration. Therefore, they were able to evaluate which machine learning algorithm they should chose for their model. However, our experimental analysis showed that the model construction type, machine learning method and input configuration shall also be taken into consideration depending on the scenario. As we have seen in our experimental analysis, our proposed methodology allowed us to draw numerous conclusions about the types of model constructions, input configurations, machine learning methods and algorithms and helped on the decision of a best combination choice for our experimental scenario.

However, this analysis also shows that not always the best combination will remain the same for all possible scenarios.

In this section we discuss how distinct scenarios may affect the model best combination choice.

We also present an analysis of potential energy-saving results where our model would be used to develop an energy-efficient management system that controls the power state of AP wireless interfaces.

A. SEASONAL INFORMATION

In our scenario, where we used Wi-Fi association information to build a wireless network energy efficient management system without real time data acquisition, month and day input features should not be used once these features showed no enhancement on the prediction model results. On the other hand, although data used in [9] and [10] present the same seasons of our experimental analysis, they showed seasonal information as a relevant input feature. Those studies were made in northern hemisphere countries in temperate regions, such as the ones found in Europe and North America, while our data were collected in a tropical country in South America. Therefore, we can conclude that seasonal information must be analyzed in these types of systems since is not always significant and depending on your building's location it should or should not be used as an input.

B. INDIVIDUAL AND COLLECTIVE COMPARISON

Another important question to answer is which type of model construction, individual or collective, should be used. Individual and collective models can have distinct results as they are trained with distinct dataset information. Our experimental analysis showed that there was no difference between the individual and collective models except for their sizes, where individual models were much smaller than the collective ones. However, it is not always true that information regarding various sensors can benefit other sensor's predictions. Also, further examination based on the scenario is required since model sizes can be relative. In our motivation scenario, for example, individual models would be actually bigger, once the collection of individual models stored at the central unit would be bigger than one single collective model capable of giving predictions for all APs. In scenarios where each individual model is deployed in its respective sensor or actuator, they would be smaller than the collective model.

C. ENERGY-SAVING ANALYSIS

From our scenario, it is possible to develop an energy-efficient management system that controls the power state of AP wireless interfaces. The developed system would use the ML model predictions to detect unoccupied APs and turn off their wireless interfaces for unoccupied periods.

We can estimate the energy saving factor of our proposed scenario using Eq. 10, presented in [35], where: P_{ext_on} and P_{ext_off} represent an AP external power source measured power in Watts (W) for cases where the wireless network interface is switched on and off respectively; and t_{on} and t_{total} represent the period of time that the APs stayed with their wireless interface switched on and total period of time that is taken into analysis respectively.

$$F = \frac{P_{ext\_on} - P_{ext\_off}}{P_{ext\_on}}(1 - \frac{t_{on}}{t_{total}}) \tag{10}$$

The result given by Eq. 10 gives the percentage of energy that can be saved from the total energy used, by switching off the AP wireless interfaces during idle time slots. Through the formula and the classifier results, it is feasible to estimate the power saving factor for the baseline proposed scenario for the month of September 2018, as follows.

From our experimental analysis using the proposed unified methodology, we selected the Col/DT/SL/APHDWD as our prediction model. Using the selected prediction model, it is possible to determine that the H building APs would stay unoccupied during 43.20% of the time for the month of September.

Through practical experiments, we measured that the consumed power values for our AP model wireless interface switched on and off states are 1,111W and 0,845W respectively. Therefore we could have saved 10.34% of the total energy consumption, if we used a mechanism as proposed in this paper for the SCIFI network in the H building during September 2018.

However, it is important to mention that the results achieved with this analysis are merely a baseline estimation. The mechanism proposed for this evaluation scenario is simple and does not take several aspects of the Wi-Fi network into consideration as others do [5], [6], [15], [16], [23], [24]. Even though, it is fair enough to assume that it gives a good baseline estimation of how much energy could have been saved using such principles with more complex and complete mechanisms, which we are going to address as future work.

VIII. CONCLUSION

In this paper, we presented a unified experimental methodology to evaluate and compare classification and regression models on their capacity to accurately predict access point demands for smart building scenarios. We conducted an experimental analysis using our proposed methodology and data collected from the UFF's SCIFI network APs, belonging to a classroom building, over a period of 6 months, from April to September 2018.

Our results show that the Col/DT/SL/APHD ML model not only achieved the best $\overline{A}$ accuracy results for the classification problem (with an $\overline{A}$ of 86.69%) but also achieved the best $\overline{RMSPE}$ results for the regression problem (with an $\overline{RMPSE}$ value of 0.29). It is also worth to notice that the mechanism proposed in this work would have saved around 10.34% of the total energy used by the SCIFI wireless network for the whole month of September 2018. Our experimental analysis showed that the proposed methodology could broadly and extensively evaluate the machine learning (ML) models. It also showed that other model parameters besides ML algorithms need to be taken into consideration when deciding the best ML model prediction to be used for smart building management systems. During our experimental analysis, we also concluded that the smart building scenario is a crucial determinant to evaluate which model parameters to choose and that, depending on the scenario, those choices might change.

Future work involves the development of a smart energy-saving mechanism for large-scale wireless networks that uses classification and regression models. Those models will use the results obtained in our experimental analysis to understand both idleness and demands of UFF's SCIFI access points. That energy-saving mechanism will operate in the SCIFI controller without requiring real-time data acquisition or high CPU power.

REFERENCES

[1] H. Zou, Y. Zhou, H. Jiang, S.-C. Chien, L. Xie, and C. J. Spanos, ''WinLight: A WiFi-based occupancy-driven lighting control system for smart building,'' *Energy Buildings*, vol. 158, pp. 924–938, Jan. 2018.

[2] D. H. Stolfi and E. Alba, ''Green swarm: Greener routes with bio-inspired techniques,'' *Appl. Soft Comput.*, vol. 71, pp. 952–963, Oct. 2018.

[3] H. Zou, H. Jiang, J. Yang, L. Xie, and C. Spanos, ''Non-intrusive occupancy sensing in commercial buildings,'' *Energy Buildings*, vol. 154, pp. 633–643, Nov. 2017.

[4] S. K. Ghai, L. V. Thanayankizil, D. P. Seetharam, and D. Chakraborty, ''Occupancy detection in commercial buildings using opportunistic context sources,'' in *Proc. IEEE Int. Conf. Pervas. Comput. Commun. Workshops*, Mar. 2012, pp. 463–466.

[5] B. Balaji, J. Xu, A. Nwokafor, R. Gupta, and Y. Agarwal, ''Sentinel: Occupancy based HVAC actuation using existing WiFi infrastructure within commercial buildings,'' in *Proc. 11th ACM Conf. Embedded Netw. Sensor Syst. - SenSys*, 2013, pp. 1–14.

[6] L. Fang, G. Xue, F. Lyu, H. Sheng, F. Zou, and M. Li, ''Intelligent large-scale AP control with remarkable energy saving in campus WiFi system,'' in *Proc. IEEE 24th Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Dec. 2018, pp. 69–76.

[7] K. Christensen, R. Melfi, B. Nordman, B. Rosenblum, and R. Viera, ''Using existing network infrastructure to estimate building occupancy and control plugged-in devices in user workspaces,'' *Int. J. Commun. Netw. Distrib. Syst.*, vol. 12, no. 1, pp. 4–29, Nov. 2014.

[8] W. Wang, J. Chen, T. Hong, and N. Zhu, ''Occupancy prediction through Markov based feedback recurrent neural network (M-FRNN) algorithm with WiFi probe technology,'' *Building Environ.*, vol. 138, pp. 160–170, Jun. 2018.

[9] F. C. Sangogboye, K. Imamovic, and M. B. Kjargaard, ''Improving occupancy presence prediction via multi-label classification,'' in *Proc. IEEE Int. Conf. Pervas. Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2016, pp. 1–6.

[10] A. Trivedi, J. Gummeson, D. Irwin, D. Ganesan, and P. Shenoy, ''ISchedule: Campus-scale HVAC scheduling via mobile WiFi monitoring,'' in *Proc. 8th Int. Conf. Future Energy Syst.*, May 2017, pp. 132–142.

[11] B. W. Hobson, D. Lowcay, H. B. Gunay, A. Ashouri, and G. R. Newsham, ''Opportunistic occupancy-count estimation using sensor fusion: A case study,'' *Building Environ.*, vol. 159, Jul. 2019, Art. no. 106154.

[12] F. Lyu, L. Fang, G. Xue, H. Xue, and M. Li, ''Large-scale full WiFi coverage: Deployment and management strategy based on user spatio-temporal association analytics,'' *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9386–9398, Dec. 2019.

[13] H. Zou, Y. Zhou, J. Yang, and C. J. Spanos, ''Towards occupant activity driven smart buildings via WiFi-enabled IoT devices and deep learning,'' *Energy Buildings*, vol. 177, pp. 12–22, Oct. 2018.

[14] F. G. Debele, N. Li, M. Meo, M. Ricca, and Y. Zhang, ''Experimenting ResourceonDemand strategies for green WLANs,'' *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 42, no. 3, pp. 61–66, Dec. 2014.

[15] K. Gomez, C. Sengul, N. Bayer, R. Riggio, T. Rasheed, and D. Miorandi, ''MORFEO: Saving energy in wireless access infrastructures,'' in *Proc. IEEE 14th Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2013, pp. 1–6.

[16] A. P. Jardosh, K. Papagiannaki, E. M. Belding, K. C. Almeroth, G. Iannaccone, and B. Vinnakota, ''Green WLANs: On-demand WLAN infrastructures,'' *Mobile Netw. Appl.*, vol. 14, no. 6, pp. 798–814, Dec. 2009.

[17] J. Lorincz, A. Capone, and M. Bogarelli, ''Energy savings in wireless access networks through optimized network management,'' in *Proc. IEEE 5th Int. Symp. Wireless Pervas. Comput.*, May 2010, pp. 449–454.

[18] W. Wang, J. Chen, and T. Hong, ''Occupancy prediction through machine learning and data fusion of environmental sensing and Wi-Fi sensing in buildings,'' *Autom. Construct.*, vol. 94, pp. 233–243, Oct. 2018.

[19] Y. Cui, X. Ma, H. Wang, I. Stojmenovic, and J. Liu, ''A survey of energy efficient wireless transmission and modeling in mobile cloud computing,'' *Mobile Netw. Appl.*, vol. 18, no. 1, pp. 148–155, Feb. 2013.

[20] S. Tang, H. Yomo, Y. Kondo, and S. Obana, ''Wake-up receiver for radio-on-demand wireless LANs,'' *EURASIP J. Wireless Commun. Netw.*, vol. 2012, no. 1, p. 42, Dec. 2012.

[21] F. Ganji, L. Budzisz, and A. Wolisz, ''Assessment of the power saving potential in dense enterprise WLANs,'' in *Proc. IEEE 24th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2013, pp. 2835–2840.

[22] M. Dalmasso, M. Meo, and D. Renga, ''Radio resource management for improving energy self-sufficiency of green mobile networks,'' *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 44, no. 2, pp. 82–87, Sep. 2016.

[23] I. Donevski, G. Vallero, and M. A. Marsan, ''Neural networks for cellular base station switching,'' in *Proc. IEEE INFOCOM Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 738–743.

[24] G. Vallero, D. Renga, M. Meo, and M. A. Marsan, ''Greener RAN operation through machine learning,'' *IEEE Trans. Netw. Service Manage.*, vol. 16, no. 3, pp. 896–908, Sep. 2019.

[25] Y. Kondo, H. Yomo, S. Tang, M. Iwai, T. Tanaka, H. Tsutsui, and S. Obana, ''Energy-efficient WLAN with on-demand AP wake-up using IEEE 802.11 frame length modulation,'' *Comput. Commun.*, vol. 35, no. 14, pp. 1725–1735, Aug. 2012.

[26] S. Pandey, D. A. Hindoliya, and R. Mod, ''Artificial neural networks for predicting indoor temperature using roof passive cooling techniques in buildings in different climatic conditions,'' *Appl. Soft Comput.*, vol. 12, no. 3, pp. 1214–1226, Mar. 2012.

[27] E. C. Goncalves, A. A. Freitas, and A. Plastino, ''A survey of genetic algorithms for multi-label classification,'' in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2018, pp. 1–8.

[28] F. Herrera, F. Charte, A. J. Rivera, and M. J. Jesus, ''Multilabel classification,'' in *Multilabel Classification*. Cham, Switzerland: Springer, 2016, pp. 17–31.

[29] E. C. Gonçalves, A. Plastino, and A. A. Freitas, ''Simpler is better: A novel genetic algorithm to induce compact multi-label chain classifiers,'' in *Proc. Annu. Conf. Genetic Evol. Comput.*, Jul. 2015, pp. 559–566.

[30] M. W. Gardner and S. R. Dorling, ''Artificial neural networks (the multilayer perceptron)—A review of applications in the atmospheric sciences,'' *Atmos. Environ.*, vol. 32, nos. 14–15, pp. 2627–2636, Aug. 1998.

[31] L. C. S. Magalhães, H. D. Balbi, C. Corrêa, R. D. T. D. Valle, and M. Stanton, ''Scifi—A software-based controller for efficient wireless networks,'' in *UbuntuNet-Connect*. Lilongwe, Malawi: UbuntuNet Alliance, 2013.

[32] F. Fainelli, ''The openwrt embedded development framework,'' in *Proc. Free Open Source Softw. Developers Eur. Meeting*, 2008, p. 106.

[33] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, ''API design for machine learning software: Experiences from the scikit-learn project,'' in *Proc. ECML PKDD Workshop, Lang. Data Mining Mach. Learn.*, 2013, pp. 1–15.

[34] W. McKinney, ''pandas: A foundational Python library for data analysis and statistics,'' *Python High Perform. Sci. Comput.*, vol. 14, no. 9, pp. 1–9, 2011.

[35] I. Haratcherev, M. Fiorito, and C. Balageas, ''Low-power sleep mode and out-of-band wake-up for indoor access points,'' in *Proc. IEEE Globecom Workshops*, Nov. 2009, pp. 1–6.

GUILHERME HENRIQUE APOSTOLO was born in Rio de Janeiro, Brazil, in 1993. He received the B.S. degree in telecommunication engineering from Universidade Federal Fluminense, Niterói, Brazil, where he is currently pursuing the M.Sc. degree in computing with the Institute of Computing. His current research interests include green networking, wireless communications, machine learning, and biomedical engineering.

FLAVIA BERNARDINI received the bachelor's degree in computer science from São Paulo State University (UNESP), the master's degree in computer science from the Institute of Mathematics and Computer Science, University of São Paulo (USP), and the Ph.D. degree from the Institute of Mathematics and Computer Science. She is currently an Associate Professor with the Institute of Computing, Fluminense Federal University (IC/UFF). She coordinates the Data Analysis Center for Citizenship (D4Ctz). She is also an Associate Researcher with the Active Design and Documentation Laboratory (ADDLabs), UFF. She has been experience in coordinating and executing research and development projects since 2007, mainly involving the development of systems that uses Artificial Intelligence, emphasizing on Machine Learning, for several problems in different application domains. In the last decade, she has turned her interests to Smart Cities theme. In this entire context, she has published several articles, with many undergraduate, master's, and doctoral degrees completed and in progress. She is also an Associate Researcher with the Brazilian Network of Intelligent and Human Cities.

LUIZ C. SCHARA MAGALHÃES received the M.Sc. degree from the Pontifícia Universidade Católica do Rio de Janeiro, in 1993, and the Ph.D. degree in computer science from the University of Illinois at Urbana Champaign, in 2002. He is currently a Professor with Universidade Federal Fluminense. His research interests include computer networks, mobility, transport protocols, loss discrimination, congestion control, and wireless networks.

DÉBORA CHRISTINA MUCHALUAT-SAADE received the bachelor's degree in computer engineering and the M.Sc. degree in computer science in 1992 and 1996, respectively, and the Ph.D. degree in computer science from the Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), in 2003. She is currently a Full Professor with the Computer Science Department, Fluminense Federal University (UFF). In 2003, she founded the MídiaCom Research Lab, UFF. She received several research grants from the Brazilian National Council for Scientific and Technological Development (CNPq), the Rio de Janeiro State's Research Support Foundation (FAPERJ), the Coordination for the Improvement of Higher Education Personnel (CAPES), and other funding agencies. Her research interests include computer networks, wireless networks, multimedia systems, and digital healthcare.