Machine Learning Based Occupancy Detection Via The Use of Smart Meters

T. Vafeiadis[1], S. Zikos[1], G. Stavropoulos[1,2], D. Ioannidis[1], S. Krinidis[1], D. Tzovaras[1] and K. Moustakas[2]

[1]Information Technologies Institute (ITI), CERTH Thessaloniki, Greece

[2]Department of Electrical and Computer Engineering, University of Patras, Patras, Greece

e-mail: {thanvaf, stavrop, szikos, djoannid, krinidis, dimitrios.tzovaras}@iti.gr; {stavrop, moustakas}@ece.upatras.gr

***Abstract* — In this work, the challenging problem of occupancy detection in a domestic environment is studied based on information gathered from electricity and water consumption smart meters. The most popular machine learning techniques, along with their boosting versions, are utilized for occupancy detection using the measurements of a door counter sensor as ground truth for training. In order to evaluate information gained from electricity and water consumption features and to reduce dataset sparsity, while maintaining the performance of classification techniques, mutual information is used as feature extraction technique. In order to determine the most efficient parameter combinations of machine learning techniques, we performed a series of Monte Carlo simulations for each method and for a wide range of parameters. Our simulation results show a superiority of Random Forest learning technique compared to the other classification techniques with accuracy slightly over 80% and F-measure with almost 84%, respectively.**

Keywords — Occupancy detection; energy data; smart meters; machine learning techniques; adaptive boosting.

I. INTRODUCTION

In the last three decades, occupancy detection in commercial or residential buildings has been researched extensively. In general, detecting when an office in a commercial building or a household is occupied, provides the ability to apply several automation applications that may contribute to the reduction of energy consumption. Available information about occupancy in a commercial or residential building can be used as input so as to intervene in buildings' energy management and performance aiming to control temperature (thermostats) automatically and efficiently, HVAC systems (Heating - Ventilation – Air Conditioning), lighting and other devices, resulting in energy conservation or increment of occupant comfort. HVAC systems and lighting are the major energy consumers in a typical office building as they consume about 70% of the distributed energy [1-3]. Another example is that in 2008 in US, buildings were responsible for 38% of CO_2 emissions, and for 71% and 39% of energy consumption and energy use, respectively [4]. Several mechanisms that are connected with estimated occupancy so as to control temperature, air-flow and lighting systems more efficiently are described in [5-7]. Information of occupancy detection can also be used by home automation systems, which are becoming very popular nowadays. Furthermore, occupancy detection and estimation in business and corporate environments can be easily utilized for extracting useful statistics and analytics about the usage of spaces as well. With regard to the spatio-temporal properties, occupancy information granularity can be classified into several categories (*Presence*, *Count*, *Localization*, *Track*, *Identity*) regarding the position and history of people in an environment [8]. Effective appliance scheduling can be achieved by applying DR (Demand Response) strategies [9] using occupancy information as one of the available inputs taken into account in the decision process.

In this work we address the challenging problem of occupancy detection (*absence – presence* or two-class classification scenario) utilizing machine learning techniques and their boosting versions on a dataset, which consists of features from energy and water consumption data received from smart meters in a domestic environment (see Section IV). In the pre-processing phase we have used a feature selection technique (Mutual Information) so as to evaluate information gained from each one of the dataset features in relation with occupancy, and to feed machine learning techniques with the most meaningful occupancy-based information. The selected machine learning techniques are Support Vector Machines, Decision Trees, Random Forest and Back-Propagation Neural Network. The boosting technique we choose in an attempt to improve classifiers' predictive performance is the popular *AdaBoost.SAMME*. The whole procedure is performed with the use of Monte Carlo simulation for different settings of internal classifiers' parameters. The main contributions of the paper are (a) the use of heterogeneous smart meters for implicit occupancy sensing, (b) the application of a feature selection step in order to determine the most important variables, (c) the evaluation of various machine learning algorithms in the occupancy detection domain using real-life data and (d) the use of the boosting technique.

The remainder of the paper is organized as follows. In Section II, we present related work and in Section III we provide a short presentation of the machine learning

techniques that were evaluated, the boosting algorithm and the evaluation criteria. The energy and water consumption dataset is described in Section IV. Mutual information as feature selection technique and its application on the dataset is described in Section V. The simulation setup and results are given in Section VI, and in section VII we draw our conclusions.

II. Related Work

Occupancy monitoring can be categorized into group-based and individual monitoring. The former refers to the estimation of the aggregated occupancy in a space, while the latter refers to the tracking of each occupant's position and identification of each occupant. Certain occupancy estimation systems require equipment such as mobile phones or radio frequency identification tags (RFID) that should be carried by the end-users. On the other hand, systems that utilize sensors such as acoustic, passive infra-red (PIR) etc. are capable of producing occupancy information without the need for users to carry a device.

As far as the use of infrastructure is concerned, explicit systems utilize occupancy sensors such as PIR motion sensors, door counters, acoustic sensors, and depth cameras to estimate occupancy, while implicit systems provide occupancy information using implicit sensing methods, such as device status and electricity consumption monitoring. The installation of occupancy sensors is not always a trivial task due to specific limitations and requirements that have to be followed. Moreover, high performance occupancy sensors that are able to count and track people, such as IR or depth cameras are of relatively high cost. In study [10], the authors utilize information received from Arduino-based wireless sensor boxes, each one placed in a different room, in order to estimate occupancy. An occupant detection, counting, and tracking system which utilizes depth cameras installed in a multi-space area, is presented in paper [11]. The authors of [12] propose an alternative method for counting the number of occupants in a conference room. The system that they developed utilizes information from chair pressure sensors, and performance evaluation showed that it is capable of providing fine-grained occupancy information.

The performance of occupancy detection and estimation usually improves when data from multiple sensors are used, as each sensor type captures different aspects of human presence. Rule-based methods and machine learning methods have been studied in the literature for estimating occupancy when fusing information from various sources. In cases where machine-learning methods are applied, occupancy estimation is regarded as a classification problem and is performed via the use of a selected classifier. Different machine-learning algorithms have been used in the literature, such as support vector machines (SVM), artificial neural networks (ANN), decision trees (DT), agent-based models and other. In the study of Yang et al. [13], learning-based methods such as a Conditional Random Field model and a Hidden Markov Support Vector Machine (HMSVM) are used, in order to estimate the number of occupants in a three-person residence by using the readings of PIR motion sensors, which were available via the alarm system. Occupancy estimation in an open-plan office is studied in [14], where information from sound, temperature, CO2, and PIR motion sensors is passed to an ANN model and the reported accuracy when estimating up to 6 occupants is up to 75%. In the work of Chaney et al. [15], an approach for analyzing data from multiple sensors, such as power meters, CO2 and temperature sensors, is introduced for estimating occupancy in a residence. The method combines the Dempster-Shafer theory with a Hidden Markov Model (HMM) and takes into account power consumption behavior, in order to compute estimations regarding participation in demand response.

An alternative approach for inferring occupancy is with the use of implicit sensing sources, such as energy data from power meters, signal strength data from mobile devices, and computer activity. Generally, the use of non-intrusive occupancy monitoring techniques is becoming more common in literature and focuses mainly on data collection from smart energy meters. In the work of Milenkovic et al. [16] the use of power meters is combined with PIR motion sensors in an effort to detect office worker activities at desk level and count the number of occupants at room level. For computer and desk work, an overall recognition accuracy of 95% was achieved and people count was estimated at 87% in the best case. Chen et al. [17] research the potential of non-intrusive occupancy monitoring by using electricity data from smart energy meters along with statistical analysis. Akbar et al. [18] used smart energy meters to count electricity consumption from several devices on employees work desks, on their research center. In that work, the use of machine learning methods along with feature extraction techniques had as a result to achieve a 94% on the harmonic mean of precision and recall (F-measure) on occupancy detection. Jin et al. [19], proposed a system for occupancy detection based on power usage, aiming at addressing the issues that arise when learning from limited or no training data.

Occupancy detection can also be achieved via the use of other sensor types, such as water meters. The amount of water consumed by occupants or electrical appliances such as dishwashers can indicate whether a building is occupied or not. Carboni et al. [20] provide an overview of water usage disaggregation systems and related techniques that are applied for water event classification in residential settings. Some water disaggregation methods are based on mono-modal sensing, while others utilize multi-modal sensing by combining different variables such as water flow, pressure etc. Recent datasets, such as the one described in [21], include information about water consumption in addition to information about electricity consumption.

III. Machine Learning algorithms – evaluation measures

In this section, we describe in short the machine learning techniques that were used in this work, such as Support Vector Machines (SVMs), Decision Trees (DT), Random Forest (RF) and Back-Propagation Network (BPN) along with the AdaBoost algorithm.

SVM is a machine learning technique based on risk minimization and it is first proposed by Boser, Guyon and Vapnik in 1992 [22]. The main objective of SVM is to construct a hyperplane as a decision boundary as the maximum margin between classified classes based on Kernel functions. In this work, we had applied two well-known Kernels, the Polynomial, denoted hereafter as SVM-POLY, and the Radial Basis Function, denoted hereafter as SVM-RBF. The SVM-RBF case relies on the Gaussian Radial Basis function kernel with its form given by: $K(x, y) = \exp(-\gamma\|x - y\|^2)$, where $\|x - y\|^2$ is the Euclidean distance between the feature vectors *x* and *y* and $\gamma = \frac{1}{2\sigma^2}$ is a positive constant, with σ being a free parameter. Along with σ, RBF kernel has another free parameter, the constant C. The SVM-POLY case relies on the Polynomial function kernel with its form given by: $K(x, y) = [x^T y + \theta]^p$, where p is the degree of the polynomial and θ is a free parameter that usually takes its values form integer space, although $\theta = 1$ is preferable as it avoids Hessian matrix to become zero. As in RBF, a free parameter C is defined.

DT is a simple and widely used classification technique. It applies a straightforward idea to solve the classification problem. DT classifier poses a series of carefully crafted questions about the attributes of the test record. Each time it receives an answer, a follow-up question is asked until a conclusion about the class label of the record is reached [23].

RF is one of the best among classification algorithms - able to classify large amounts of data with accuracy. RF are an ensemble learning method (one can think that RF is a form of n - nearest neighbor predictor) for classification that construct a number of decision trees at training time and outputting the class that is the mode of the classes output by individual trees [24].

Artificial Neural Networks (ANNs) have seen an increasing interest over the last few years and are being successfully applied on a significant range of problem domains. One of the most popular ANNs algorithms is the back propagation algorithm. The back propagation algorithm looks for the minimum of the error function in weight space using the method of gradient descent. The combination of weights which minimizes the error function is considered to be a solution of the learning problem [25].

The AdaBoost algorithm is now a well-known and deeply studied method to build ensembles of classifiers with very good performance [26]. The algorithm learns a set of classifiers, also known as weak learners, in order to produce the final, stronger classifier. The weak learners are obtained sequentially, using re-weighted versions of the training data, with the weights depending on the accuracy of the previous classifiers. The training set is always the same at every iteration, with each training instance weighted according to its (mis) classification by the previous classifiers.

For the two-class classification scenario, in order to assess our models, we use the measures of precision, recall, accuracy and F-measure, which are computed from the contents of the confusion matrix of the classification predictions (see Table I). True positive and false positive cases are denoted as TP and FP, while true negative and false negative are denoted as TN and FN respectively. In order to fit the classification evaluation in occupancy detection problem, we will assign the classes of absence and presence.

Precision is the ratio of predicted true positive cases to the sum of true positives and false positives and is given by the equation:

$$Precision = TP / (TP+FP) \quad (1)$$

Recall is the proportion of the true positive cases to the sum of true positives and false negatives and is given by the equation:

$$Recall = TP / (TP+FN) \quad (2)$$

Accuracy is the fraction of the total number of predictions that were correct.

$$Accuracy = (TP+TN) / (TP+FP+TN+FN) \quad (3)$$

Precision or recall alone cannot describe a classifier's efficiency. Therefore, F-measure is introduced as a combination of these two metrics. It is defined as twice the harmonic mean of precision and recall, and is the metric we will be most referring to.

$$F\text{-}measure = (2*Precision*Recall)/(Precision+Recall) \quad (4)$$

A value closer to one, means better combined precision and recall of the classifier, whereas lower values imply worst accuracy or precision or both.

TABLE I. CONFUSION MATRICES FOR OCCUPANCY DETECTION

		Predicted class	
		Absence	**Presence**
Actual Class	**Absence**	*TP*	*FN*
	Presence	*FP*	*TN*

IV. DESCRIPTION OF THE DATASET

Three different systems were deployed in a domestic environment which accommodates 3 occupants, in order to capture and store the energy, water, and occupancy related data. The data collection was performed for about 1 month, during July 2016. Power consumption and occupancy data are stored in a MySQL database deployed locally. On the contrary, water consumption data are sent over the Internet and stored in the cloud. More details about data storage and dataset generation are presented next.

A. Energy Consumption Dataset

The power consumption of crucial electrical appliances is monitored via a wireless network of smart plugs that utilize the ZigBee protocol (https://www.plugwise.com). The installed smart plug modules communicate with each other forming a network of mesh topology. Each one of the following appliances was attached to a smart plug: TV, washing machine, refrigerator, and hair dryer. Furthermore, a special built-in module, which is connected to the wiring, is

used in order to monitor the power consumption of the electrical kitchen appliance. An aggregator application has been developed and installed on a PC. It requests the current power consumption from each module every minute, receives the corresponding messages which include the measured power consumption of the connected appliance in Watts, and then stores the data directly into the database. It is worth noting that the timestamps of all events that are generated by the three systems are in UTC.

B. Water Consumption Dataset

Four water consumption sensors have been installed for monitoring and logging of the usage of water by the occupants. The monitored supplies are the cold water and the hot water supplies in the kitchen, the water consumed by the dishwasher and the water consumed by the washing machine.

C. Occupancy Dataset

A highly accurate, active infrared door counter sensor which includes four receivers has been installed at the main entrance of the house. The sensor is able to detect entries and exits based on the break-reset timings of the beams. Thus, it keeps track of the number of occupants that are present in the residence and dispatches the occupancy value to a web service, on change. The web service that has been developed is running locally and stores the received occupancy events along with the timestamp into the database.

D. Data pre-processing

After retrieving the raw data of the three systems, a processing step was performed in order to create the final aggregated dataset which includes events per 1-minute intervals of all the measured features. Periods during which the occupants were on vacation were excluded from the final dataset. Moreover, periods in which some of the data were not available due to power failures or network connectivity issues, were excluded as well.

The initial aggregated dataset constructed after processing the raw data contains 9 features [Central Power (lights of the domestic environment), Refrigerator, TV, Washing Machine, Dryer, Cold Water - Kitchen, Hot Water - Kitchen, Dishwasher - Water, Washing Machine - Water] denoted hereafter as [CP, R, TV, WM, D, CWK, HWK, DW, WMW] and the target Occupancy, denoted hereafter as [OCCUP]. The dataset contains energy and water consumption data of 1-minute resolution for a time interval of 16 consecutive days during summer time. Thus, the shape of overall dataset is 23040x9 (without taking into account the target feature) and its sparsity is 74.44%. Fig. 1 depicts the energy consumption of some electrical devices, with regard to occupancy.

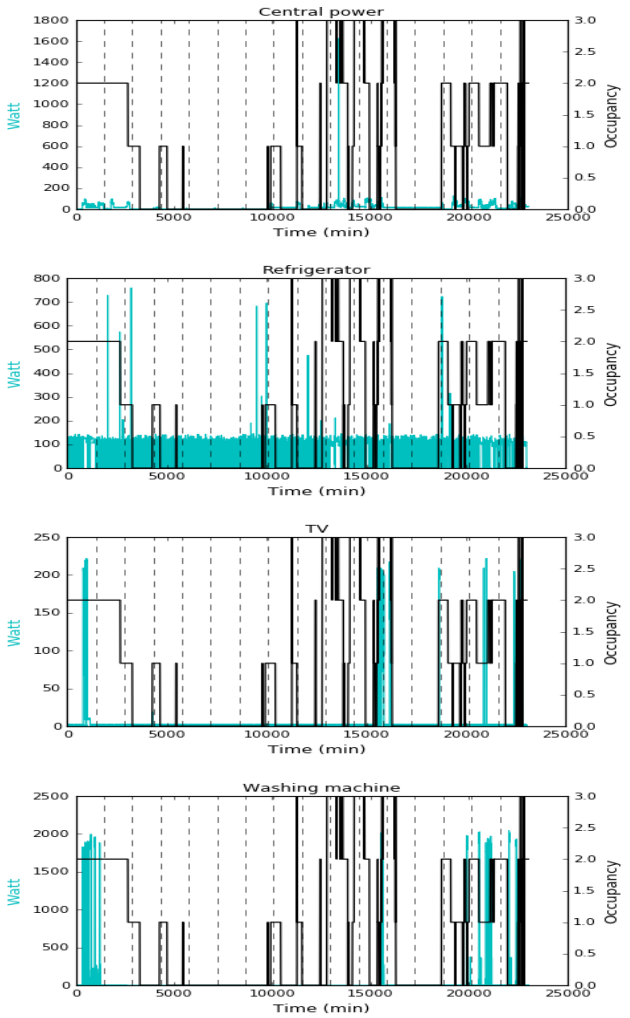

Figure 1. Energy consumption of central power, refrigerator, TV and washing machine electrical devices (cyan solid line - left axis) with regard to occupancy (black solid line - right axis).

One can see from Figure 1 that Central Power, TV and Washing Machine features are somehow correlated with occupancy. On the other hand, the calculation of Pearson's correlation among dataset features (not presented here) does not reveal clearly such a behavior.

V. MUTUAL INFORMATION

We all acknowledge the fact that interdependency exists among human habits and occupancy inference. In our work, these habits are denoted by the 9 features described previously. Thus, in order to rank the influence of each feature to occupancy inference and extract the more useful information, we have used Mutual Information (MI) as the feature selection technique. MI measures how much one random variable provides information about another. It is a dimensionless quantity, and can be thought of as the reduction in uncertainty about one random variable given knowledge of another. High mutual information indicates a large reduction in uncertainty; low mutual information

indicates a small reduction; and zero mutual information between two random variables means the variables are independent.

In Table II, the features with the highest MI, based on occupancy, are presented in descending order. According to these results, intuitively, we decide to use only the top-5 ranked features for occupancy inference, meaning Central Power, Cold Water – Kitchen, Washing Machine, Refrigerator and Washing Machine – Water. Under this condition, the shape of overall dataset is 23040x5 (again without taking into account the target feature) and its sparsity is reduced to 70.76% (from 74.44%).

TABLE II. DATASET'S FEATURES IN DESCENDING ORDER OF IMPORTANCE

Features ranking
Central Power
Cold Water - Kitchen
Washing Machine
Refrigerator
Washing Machine - Water
TV
Hot Water - Kitchen
Dishwasher - Water
Dryer

In order to avoid mixing test cases, in section of experimental results presented afterwards, the initial dataset (23040x9) will be denoted hereafter as *Initial-DS* and the dataset emerged after the use of MI (23040x9) will be denoted hereafter as *MI-DS.*

VI. EXPERIMENTAL RESULTS

A. Simulation setup

Our main objective is to find the predictive model that is more efficient on occupancy inference based on energy and water consumption data. To that end, our simulation schema is based on the application of all tested classifiers and their boosting versions on both *Initial-DS* and *MI-DS*. For cross-validation of our results, we generate a training set and a testing set, in a percent of 70% and 30%, respectively of the tested dataset. We generate 100 Monte Carlo iterations for different parameter scenarios in each classifier. The tested classifiers were SVM, DT, RF and BPN models. We employed the boosting technique with 5 weak learners (refer to Section III) to all the classifiers mentioned above in order to improve classification performance. For SVM-POLY, θ takes the values $\theta = (start = 30, end = 60, step = 6)$ and the polynomial degree takes the values $p = (2,7,1)$. For SVM-RBF σ varies same as θ and the constant C as $C = (1000,10000,2000)$. The classic BPN has a single hidden layer and the number of neurons varies as $n = (100,200,20)$. The RF has an ensemble of $estimators = (20,100,20)$ DTs. The combination of all values of parameters and a size of 100 Monte Carlo iterations for each case, results in an overall of 6100 tested cases.

B. Simulation results

Due to the vast amount of simulation results, we present results from specific simulation cases on the *Initial-DS* and from the application of AdaBoost on both *Initial-DS* and *MI-DS*.

TABLE III. PRECISION, RECALL, ACCURACY AND F-MEASURE (ESTIMATED AVERAGES) FOR 100 MONTE-CARLO ITERATIONS FOR RF AND BPN CLASSIFIERS, ON *INITIAL-DS*.

Classifier: RF				
Estimators	**Precision (%)**	**Recall (%)**	**Accuracy (%)**	**F-measure (%)**
20	72.78	85.94	78.24	78.81
40	73.03	86.31	78.33	79.11
60	73.17	86.42	**79.02**	79.24
80	73.44	86.34	78.83	79.37
100	73.56	86.49	78.75	79.50
Classifier: BPN				
Neurons	**Precision (%)**	**Recall (%)**	**Accuracy (%)**	**F-measure (%)**
100	74.79	83.28	76.24	78.81
120	75.93	84.31	75.87	79.90
140	75.21	83.33	**76.71**	79.06
160	76.07	84.52	76.33	80.07
180	76.12	84.03	76.58	79.88

Table III shows the simulation results of RF and BPN, when applied on the *Initial-DS*, and one can see that the highest accuracy, when RF is applied, is 79.02% (highlighted value on RF case) ($estimators = 60$) and 76.71% when BPN is applied ($neurons = 120$) (highlighted value on BPN case). On the other hand, SVM-POLY achieved an accuracy of 76.45% ($p = 2, \theta = 42$), while SVM-RBF achieved an accuracy of 74.82% ($C = 5000, \theta = 36$). Preliminary simulations results show a superiority of RF and

DT classifiers compare to other for our classification scenario. RF classifier had achieved 80% for accuracy and 83% for F-measure (on average of 100 Monte-Carlo iterations), with DT to follow very closely. The rest of the tested classifiers achieved below 78% on accuracy and around 80% for F-measure. Tables IV and V present precision, recall, accuracy and F-measure (on average of 100 Monte-Carlo iterations) with the application of boosting on tested classifiers on both *Initial-DS* and *MI-DS*, respectively.

TABLE IV. PRECISION, RECALL, ACCURACY AND F-MEASURE (ESTIMATED AVERAGES) FOR 100 MONTE-CARLO ITERATIONS WITH THE APPLICATION OF ADABOOST, ON *INITIAL-DS*.

Classifier	Parameters of weak learners	Precision (%)	Recall (%)	Accuracy (%)	F-measure (%)
SVM – POLY	p = 2, 3, 4, 5, 6 θ = 30, 36, 42, 48, 56	74.73	90.01	79.31	81.66
SVM – RBF	C = 1000, 3000, 5000, 7000, 9000 θ = 30, 36, 42, 48, 56	74.42	87.76	78.95	80.54
DT	-	74.03	93.89	80.14	82.79
RF	20, 40, 60, 80, 100	74.44	94.74	**80.53**	**83.37**
BPN	n = 100, 120, 140, 160, 180	73.75	93.20	79.90	82.34

TABLE V. PRECISION, RECALL, ACCURACY AND F-MEASURE (ESTIMATED AVERAGES) FOR 100 MONTE-CARLO ITERATIONS WITH THE APPLICATION OF ADABOOST, ON *MI-DS*.

Classifier	Parameters of weak learners	Precision (%)	Recall (%)	Accuracy (%)	F-measure (%)
SVM – POLY	p = 2, 3, 4, 5, 6 θ = 30, 36, 42, 48, 56	74.79	89.34	79.83	81.42
SVM – RBF	C = 1000, 3000, 5000, 7000, 9000 θ = 30, 36, 42, 48, 56	74.35	89.07	80.06	81.04
DT	-	74.89	91.37	**80.94**	82.31
RF	20, 40, 60, 80, 100	73.91	95.17	80.23	**83.20**
BPN	n = 100, 120, 140, 160, 180	74.01	92.83	80.21	82.36

From Table IV, one can see that the RF with AdaBoost achieves the higher performance on the classification scenario (absence – presence), compared to the other tested classifiers (see highlighted values), with 80.53% accuracy (83.37% F-measure). The second best classifier is DT with 80.14% accuracy (82.79% F-measure). The application of boosting provides an improvement of maximum 2% of the performance of the RF classifier compared to the non-boosted case of RF for the best set of simulation parameters. This outcome, point to the fact that the application of boosting saved us time and effort to search for the appropriate set of simulation parameters, in order to achieve the best predictive performance of the classifier. Same holds when feature selection based on MI is applied (see Table V), where simulation results are as much the same with the previous case (compare Tables IV and V).

From Table V, one can see that the DT with AdaBoost achieves the higher performance compared to the other tested classifiers (see highlighted values) with 80.94% accuracy (82.31% F-measure), while the RF follows closely in accuracy (80.23%), but achieves higher F-measure compared to DT (83.20%).

VII. CONCLUSIONS

The purpose of this work was to shed some light on the challenging problem of occupancy detection (absence – presence) on a residential environment, utilizing well known machine learning techniques and their boosting version, on data gathered from smart meters (energy and water consumption of the household). Cross-validation and Monte Carlo simulations where used in an effort to find machine learning techniques that are effective and efficient on this classification problem. Moreover, Mutual Information is used for feature selection in an effort to reduce dataset's sparsity and to retain classification performance. Feature selection technique managed to reduce dataset's sparsity around 4%. As for classification performance, Random Forest and Decision Tree classifiers under their boosting versions, show a slightly higher accuracy compared to the other tested classifiers, but they have managed to achieve an overall great performance (F-measure: 83.37% and 82.79%, respectively).

Our simulation results show that machine learning techniques have great potentials to face the challenging problem of occupancy detection. In future work, our plans are to focus more on the dataset and the features used as input to the classifiers, by taking into account different features and using other feature selection techniques, in an effort to find a more robust predictive model.

ACKNOWLEDGMENT

This work has been partially supported by the European Commission through the project HORIZON 2020-INNOVATION ACTIONS (IA)-723059-enCOMPASS.

REFERENCES

[1] L. Pérez-Lombard, J. Ortiz, and C. Pout, "A review on buildings energy consumption information", Energy and Buildings, vol. 40, no. 3, pp. 394–398, 2008.

[2] T.A. Nguyen, and M. Aiello, "Energy intelligent buildings based on user activity: a survey", Energy and Buildings, vol. 56, pp. 244–257, 2013.

[3] C. Spataru, and S. Gauthier, "How to monitor people "smartly" to help reducing energy consumption in buildings?", Architectural Engineering and Design Management, vol. 10, no. 1-2, pp. 60–78, 2014.

[4] U. S. D. of Energy (DOE), The 2008 building energy databook, Washington, D.C., 2008.

[5] J. Lu, T. Sookoor, V. Srinivasan, G. Gao, B. Holben, J. Stankovic, E. Field, and K. Whitehouse, "The smart thermostat: Using occupancy sensors to save energy in homes", Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems, 2010, DOI: 10.1145/1869983.1870005.

[6] J. Scott, A. Bernheim Brush, J. Krumm, B. Meyers, M. Hazas, S. Hodges, and N. Villar. "Preheat: Controlling home heating using occupancy prediction", Proceedings of the 13th international conference on Ubiquitous computing, 2011, DOI: 10.1145/2030112.2030151.

[7] X. Guo, D. Tiller, G. Henze, and C. Waters, "The performance of occupancy-based lighting control systems: A review", Lighting Research and Technology, vol. 42, no. 4, pp. 415-431, 2010.

[8] T. Teixeira, G. Dublon, and A. Savvides, "A Survey of Human-sensing: Methods for Detecting Presence, Count, Location, Track, and Identity", ENALAB Technical Report, vol. 1, no. 1, 2010.

[9] B. Yuce, Y. Rezgui, and M. Mourshed, "ANN–GA smart appliance scheduling for optimised energy management in the domestic sector", Energy and Buildings, vol. 111, pp. 311-325, 2016.

[10] Z. Yang, and B. Becerik-Gerber, "Cross-space building occupancy modeling by contextual information based learning", Proceedings of the 2nd International Conference On Embedded Systems for Energy-Efficient Built Environments, ACM, pp. 177-186, 2015.

[11] S. Krinidis, G. Stavropoulos, D. Ioannidis and D. Tzovaras, "A Robust and Real-Time Multi-Space Occupancy Extraction System Exploiting Privacy-Preserving Sensors", Proceedings of International Symposium on Communication Control and Signal Processing, pp. 542-545, 2014.

[12] T.M. Labeodan, W. Zeiler, G. Boxem, and Y. Zhao, "Occupancy measurement in commercial office buildings for demand-driven control applications", Energy and Buildings, vol. 93, pp. 303-314, 2015.

[13] L. Yang, K. Ting, and M.B. Srivastava, "Inferring Occupancy from opportunistically Available Sensor Data", IEEE International Conference on Pervasive Computing and Communications, pp. 60–68, 2014.

[14] T. Ekwevugbe, N. Brown, V. Pakka, and D. Fan, "Real-time building occupancy sensing using neural-network based sensor network", IEEE International Conference on Digital Ecosystems and Technologies, 2013, DOI: 10.1109/DEST.2013.6611339.

[15] J. Chaney, E.H. Owens, and A.D. Peacock, "An evidence based approach to determining residential occupancy and its role in demand response management", Energy and Buildings, vol. 125, pp. 254-266, 2016.

[16] M. Milenkovic, and O. Amft, "An opportunistic activity-sensing approach to save energy in office buildings", Proceedings of the fourth international conference on Future energy systems, ACM, pp. 247-258, 2013.

[17] D. Chen, S. Barker, A. Subbaswamy, D. Irwin, and P. Shenoy, "Non-Intrusive Occupancy Monitoring using Smart Meters", Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings, 2013, DOI: 10.1145/2528282.2528294.

[18] A. Akbar, M. Nati, F. Carrez, and K. Moessner, "Contextual Occupancy Detection for Smart Office by Pattern Recognition of Electricity Consumption Data", IEEE International Conference on Communications, pp. 561-566, 2015.

[19] M. Jin, R. Jia, and C. Spanos, "Virtual occupancy sensing: Using smart meters to indicate your presence", *IEEE Transactions on Mobile Computing*, 2017.

[20] D. Carboni, A. Gluhak, J.A. McCann, and T.H. Beach, "Contextualising Water Use in Residential Settings: A Survey of Non-Intrusive Techniques and Approaches", Sensors, vol. 16, no. 5, pp. 738-745, 2016.

[21] S. Makonin, B. Ellert, I.V. Bajić, and F. Popowich, "Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014", Scientific data, vol. 3, 2016.

[22] B.E. Boser, I.M. Guyon, and V.N. Vapnik, "A training algorithm for optimal margin classifiers", Proceedings of the Fifth Annual Workshop on Computational Learning Theory, ACM, pp. 144–152, 1992.

[23] S.J. Lee, and K. Siau, "A review of data mining techniques", Industrial Management & Data Systems, vol. 101, no.1, pp. 41–46, 2001.

[24] L. Breiman, Random forests – random features. Technical Report 567, Statistics Department, University of California, Berkeley, ftp://ftp.stat.berkeley.edu/pub/users/ breiman.

[25] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning representations by back-propagating errors", Nature, vol. 323, pp. 533-536, 1986.

[26] T. Hastie, R. Tibshirani, and J. Friedman. The Elements of Statistical Learning. Springer, 2nd edition, 2001.