Comparison of building occupants activities detection methods using simulated data: a case of window opening detection

Chuhao Jiang*[1], Marie-Lise Pannier[1], Alain Godon[2], David Bigaud[1]

[1] Univ Angers, LARIS, SFR MATHSTIC, F-49000 Angers, France
[2] Univ Angers, Département Systèmes Automatisés et Génie Informatique, Polytech Angers, F-49000 Angers, France
***chuhao.jiang@etud.univ-angers.fr**

RÉSUMÉ. Une bonne connaissance des usages et des usagers des bâtiments est nécessaire au développement de stratégies de gestion optimales tant sur l'énergie que sur le confort. Toutefois, caractériser les habitudes des occupants n'est pas une tâche simple. L'analyse des nombreuses données collectées dans les bâtiments connectés représente un atout pour mieux comprendre l'occupation. Préalablement à la détection d'actions adaptatives à partir de mesures de capteurs connectés, la présente étude s'attache à comparer les performances de méthodes de détection sur la base de données simulées. Il s'agit ici d'identifier des ouvertures de fenêtres dans une salle de classe en analysant des résultats de simulations thermiques dynamiques. L'approche d'apprentissage automatique supervisée offre de meilleures performances de détection dans ce cas d'étude théorique. Cependant, l'approche semi-supervisée semble la plus adaptée en vue d'une transposition à des données mesurées in situ, où l'état d'ouverture des fenêtres ne sera pas forcément connu à chaque observation du jeu d'entrainement.

MOTS-CLÉS : Bâtiments connectés, Simulation, Détection d'activités des occupants.

ABSTRACT. A good knowledge of the building uses and users is a prerequisite to develop management strategies that are optimal for both energy and comfort management. However, characterising the habits of occupants is a challenging task. The analysis of the large amount of data collected in connected buildings is an opportunity to better understand the occupancy. Prior to the detection of adaptive actions based on measurements from connected ambient sensors, the present study aims to compare the performance of detection methods from simulated data. The goal is to detect window openings in a classroom by analysing the results of dynamic building energy simulations. If the supervised machine learning approach offers better detection performances in this case of study, the semi-supervised approach seems more proper for a transposition to measured data, as the window opening state is not necessarily known at each observation of the training set.

KEYWORDS : Smart building, Simulation, Occupant's activity detection.

1. INTRODUCTION

It is crucial to improve the performance of buildings to achieve a sustainable energy society. By using Building Management Systems (BMS), we aim to minimize the energy consumption in buildings and to maximise occupants' comfort. However, to achieve a user centric energy management, information related to the occupants' behaviour such as window openings and occupants' activities are required as an input of the BMS. Several methods have been proposed in the literature to detect and classify user's events (Xilei *et al.* 2020). The most recent used methods for occupants' activity or presence detection are deep learning methods such as artificial or recurrent neural networks (ANN and RNN) (Xilei *et al.* 2020). As an example, Danaei-Mehr *et al.* (2016) used three different ANN methods

to recognize detailed human activities such as bathing, grooming and toileting and Paradiso *et al.* (2013) used an ANN to detect the power of appliance, for eight monitored devices. Regarding the window opening detection, the logistic regression, borrowed by machine learning from statistics, is by far the most frequently applied method (Xilei *et al.* 2020). For instance, Stazi *et al.* (2017) applied a logistic and a linear regression model to detect the windows status in classrooms.

It turns out that supervised methods are the most used in the literature at the expense of unsupervised methods. For supervised methods, the data are labelled. This means that the class of the data, e.g., opened or closed in the case of windows state detection, is known. However, in practical applications, the label could be unknown as labelling data is resource consuming and can be intrusive: it requires for instance to have many sensors available to detect the presence of occupants or windows openings. Another approach to achieve the goal of detecting occupants' behaviour with a fewer labelling effort is to use semi-supervised methods, such as the ones applied in anomaly detection problems. Such methods are useful to detect outliers, i.e., points that differs significantly from a known or a normal state, from partially labelled data. Choi and Yoon (2021) used a semi-supervised method base on AutoEncoder (AE) in order to detect and to diagnose faults on a district heating substation.

In this article, different methods (supervised, unsupervised and semi-supervised ones) are investigated. Their performances are compared based on their ability to detect windows openings in a simulated case study.

2. PROJECT BACKGROUND

Within the BIoT project (Building Internet of Things – https://biot.u-angers.fr/), two lecture rooms of the University of Angers have been selected to assess the ability to detect events caused by occupants. Indoor ambient sensors and windows openings sensors have recently been installed inside. Measurements are currently being recorded, and the first analysis of the events (windows openings) detection ability presented in this article is performed using simulations carried out on a similar lecture room modelled within the software DesignBuilder. Weather file for the Nantes (a city near Angers) area has been linked to the simulation. The simulated room has north-facing windows as in reality. The external walls have a U-value of 0.350 $W/m^2/K$. The simulation period is one year from first of January with 15 minutes time step. Because the data is virtual, it is possible to construct a data with extreme conditions to verify the ability of the model: to detect arbitrary irregular data and to handle highly imbalance data. Therefore, window opening schedules have been randomly generated with Python by allowing only a few openings during the year. Some constraints have been defined: window opening can only occur between 8 a.m. to 7 p.m. on the weekdays, window status can change every 15 minutes, and window remains open up to 30 minutes. In the project, window opening is a categorical variable: opening is recorded as 1 and closing as 0. The window status is highly imbalanced in the generated date: there are only 147 window opening cases among the 35,040 simulated time steps. The window opening rate in whole datasheet is about 0.0042.

There are totally 13 different features (or variables) in the time series in the simulation Based on the research purpose, features which are, to our knowledge, more correlated to the window opening and occupants presence are selected. So seven useful features time series are followed in the simulation. It includes one categorical variable (window status) and six continuous variables: computers heat gains, occupants heat gain, solar gain from window, heat consumption, and indoor and outdoor temperatures.

In Figure 1, the Pearson correlation coefficient between each feature can be seen. A lighter colour indicates a higher correlation. The correlation with the feature to predict ("opening") is discussed here. As expected, the window status is mainly correlated to the heating consumption: when the windows are opened, heaters need to maintain the heat setpoint, so the heat consumption increases at that time step.

3. METHODOLOGY OF ACTIVITIES DETECTION

To compare the performance of the three approaches, the methodology presented in Figure 2 is followed. For the fairness of the comparison, the most common and classical algorithms are selected in these three approaches. Principal Component Analysis (PCA)， logistic regression (LR) and AE are the very representative algorithms in three approaches respectively. Multivariate Gaussian (MG) distribution can be considered as one of the simplest algorithms to be used as a comparison to judge the performance of other algorithms.

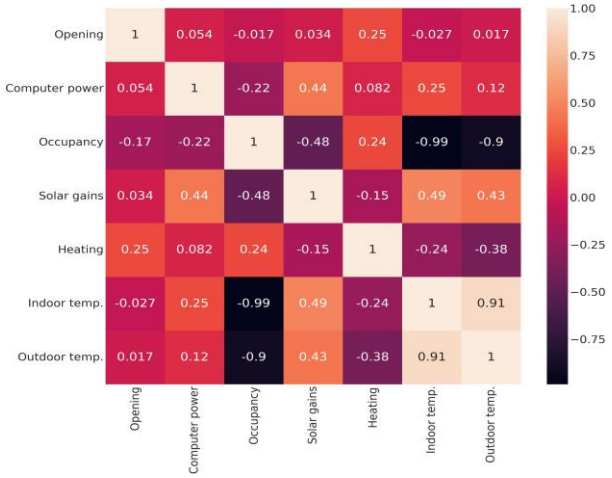

Figure 1 : Correlation between the selected features.

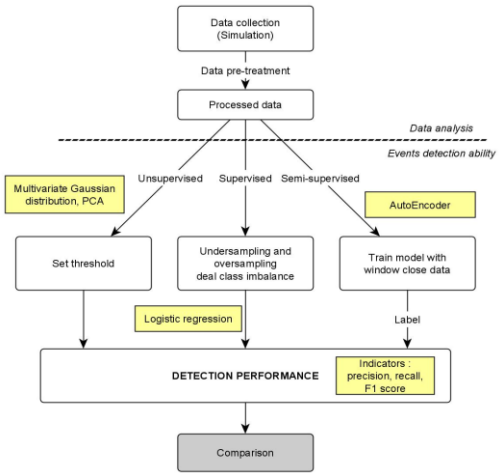

Figure 2 : Methodological steps.

3.1. DATA PRE-TREATEMENT

All-time series represent 35,040 simulated observations of seven variables, but time series is not considered as an input in this paper. Observations are randomly split into two parts. 80 % are used for training and 20 % for testing. For the unsupervised methods, the windows status is neither used in the training set nor in the testing one. The supervised method processes all seven variables in the training set. The windows status variable is removed from the testing set since it must be considered here as an output. As far semi-supervised method is concerned, observations corresponding to openings are filtered out from the training set, so that the training set contains only closing state (i.e., normal or reference state) observations. However, the testing set contains all data corresponding to both openings and closings (to be compared to the reference state).

3.2. EVENT DETECTION METHODS

3.2.1. Unsupervised method

Two unsupervised methods are applied to the pre-processed data based on the use of a MG distribution and on the PCA.

3.2.1.1. Multivariate Gaussian distribution

In the unsupervised method, the most straightforward way is to estimate the density of the data and to set a density threshold. Anything below this threshold is considered as an outlier. In order to take

correlations into account the MG distribution (Bishop 1994) has been used in this project. When the dataset has m features $\boldsymbol{x} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$, and each feature includes n data $x^{(i)} \in \mathbb{R}^{n\times 1}$, then the MG distribution density can be expressed as follow:

$$p_X\left(x^{(1)}, \dots, x^{(m)}\right) = \frac{exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)}{\sqrt{(2\pi)^n|\boldsymbol{\Sigma}|}} \quad (1)$$

With $\boldsymbol{x}$ is a real (n * m)-dimensional matrix, $\boldsymbol{\mu}$ a (n * m)-dimensional matrix of mean values and $|\boldsymbol{\Sigma}| \equiv det\boldsymbol{\Sigma}$ is the determinant of $\boldsymbol{\Sigma}$, the (n * n)-dimensional covariance matrix.

When using the MG distribution for the anomaly detection, a threshold ϵ is set by expert knowledge depending on the context. A sample x is consider as a normal point when $p(x) > \epsilon$, otherwise the sample is considered as an outlier.

3.2.1.2. Principal Component Analysis

Another approach for activity detection within unsupervised methods is data reconstruction, relying on a dimension reduction from the initial number of features to a new set of transformed features with lower dimension. After the reconstruction, if a single point does not show the same characteristics as the overall data sample, such as deviating from other data sample, it may show that the data sample is an outlier. PCA (Duda 2000) is a linear dimensionality reduction method. It provides a set of orthogonal eigenvectors of data representation in the minimum mean square sense by linear combination of data features to capture the maximum change direction of the data. The eigenvector corresponding to the large eigenvalue is its principal component. The first K (sorted in descending order of their values) eigenvectors form a matrix P_K. For the data sample x, the error threshold can be set for anomaly detection applications as:

$$E_{PCA}(x) = \|x - (P_K(P_K^T P_K)^{-1}P_K^T)x\|_2 = \|x - P_K P_K^T\|_2 \quad (2)$$

The matrix of the new data set reconstructed after PCA dimensionality reduction is $x1$. For the anomaly score of each point in the data sample x is:

$$Score(x) = \sum_{k=1}^{n}(|x - x1|) \times ev(K) \quad (3)$$

$$\text{with } ev(K) = \frac{\sum_{j=1}^{K}\gamma_j}{\sum_{j=1}^{n}\gamma_j} \quad (4)$$

The $ev(K)$ is cumulated ratio of variance explained by the first K principal component factors.

3.2.2. Supervised method

LR algorithm (Cox 1958) is used to detect window opening. It is a nonlinear regression model. Based on a known data set, a nonlinear function is identified that fits the data as much as possible.

The biggest challenge when using supervised method for activity detection is class-imbalance (between normal and abnormal classes). It occurs for fault detection because the number of anomalies is rare compared to normal sample data. In this case, it is difficult to learn effective rules from a supervised model with a limited number of negative samples (outlier). Usually, classification machine learning tasks expect the samples size of each category to be balanced. In our present study, the classes are strongly (and deliberately) imbalanced.

The results are first presented without handling imbalanced data. Then, two processing methods for class-imbalance are adopted. Firstly, undersampling (Liu, Wu, and Zhou 2009), it is the process of selecting a small number of samples at random from the majority class (in our study, the closed window state class). This small sample is then merged with the original minority class samples into a new training data set. Secondly, oversampling, it is the process of expanding the minority class by simply copying observations from this class. The random oversampling algorithm is the simplest method. However, oversampling approaches may lead to the problem of model overfitting, as it makes the information learned by the model too specific. One way to improve this method consists in using data synthesis to generate more data based on existing data. Synthetic minority oversampling technique (SMOTE) (Chawla *et al.* 2002) is used in the present study.

In terms of a partial conclusion on supervised and unsupervised methods, the unsupervised method has the advantage that it does not require labelled data and therefore has a wide range of applications. On the other hand, it requires an assumption about the distribution of the data. In contrast, the supervised method is more accurate, but requires a lot of labelled data, which is a waste of resources. A good compromise is therefore an intermediate method: semi-supervised.

3.2.3. Semi-supervised method

There is an observable trend toward semi-supervised learning-based modelling, which is a method that combines supervised and unsupervised learning. It uses partial labelled data for training, and large number of unlabelled data for classification. This approach seems promising for our case study.

AE-based methods are particularly studied and appreciated to meet these objectives. These methods (Bourlard and Kamp 1988) essentially use a neural network to produce a low dimensional representation of a high dimensional input. AE consists in two main modules, the encoding process and the decoding process. The encoder maps the input sample ***x*** to the feature space ***z*** in the encoding process. Then the decoder maps the abstract feature ***z*** back to the original space to obtain the reconstructed sample ***x1*** in the decoding process. The optimization goal is to optimize both encoder and decoder by minimizing the reconstruction error, so as to learn the abstract feature ***z*** for the sample input ***x***.

3.3. EVENT DETECTION PERFORMANCE QUANTIFICATION

Predicted and true openings are confronted using a confusion matrix as in Table 1. Correctly predicted states are on a green background. Based on the confusion metric, additional indicators are computed to quantify the performance of each method, namely precision and recall. The precision is the ratio of predicted true openings (TP) on all predicted openings (TP+FP). The recall is the ratio of predicted true openings (TP) on all true openings (TP+FN). In this application, the aim is to detect all true window openings. The recall must therefore be maximised.

		Predicted	
	Confusion matrix	Window close	Window open
Reality	Window close	True negative (TN)	False positive (FP)
	Window open	False negative (FN)	True positive (TP)

Table 1 : Shape of a confusion matrix.

4. RESULTS

The results of the different methods are presented below, and a comparison of the results can be found at the end of this section.

For unsupervised learning with the use of the MG distribution (see §3.2.1.1), four different threshold ϵ numbers are tested. The number that will give the best precision and recall will be chosen as the threshold. In Table 2 the window opening detection rate (i.e. recall) is 88% $\left(\frac{64}{64+9}\right)$. In the window opening prediction, all 9 prediction errors occurred in March.

		Predicted	
	Confusion matrix	Window close	Window open
Reality	Window close	6 843	135
	Window open	9	64

Table 2 : Confusion matrix for MG distribution.

For unsupervised learning relying on PCA data pre-treatment (see §3.2.1.2), the original dimension corresponding to the six features (seven features but we remove windows status) has been reduced to two. The reconstruction score is calculated and it turns out that it is low for most samples. Normally, the threshold of the reconstruction score is set by using the domain expertise, since it directly impacts the precision and recall. By looking at the distribution of reconstruction scores, the range of threshold can be roughly determined Here, the mean plus standard deviation is a reasonable threshold. From the Table 3 for this unsupervised method mixed with PCA, the model has a recall of 81%. Considering this is a highly imbalanced datasheet, this is a good result. In the window opening prediction, 20 of the 22 prediction errors occurred between September and April period.

		Predicted	
	Confusion matrix	Window close	Window open
Reality	Window close	26 668	1 246
	Window open	22	96

Table 3 : Confusion matrix for PCA.

In Table ,Table and Table 6, we propose the results obtained by using supervised learning methods (see §3.2.2) with unprocessed imbalance datasheet, undersampling and oversampling techniques processed datasheet. For LR with the unprocessed imbalance datasheet (Table 4), the result shows the algorithm can't detect the minority class with a perfect performance due to the highly imbalance datasheet, the recall equals 73%. In contrast, undersampling and oversampling techniques can handle the problem very well. It must be noted that both methods show perfect results.

		Predicted	
	Confusion matrix	Window close	Window open
Reality	Window close	10474	0
	Window open	10	28

Table 4 : Confusion matrix for LR without undersampling and oversampling.

For LR with undersampling (Table 5), the recall equals 100%.

		Predicted	
	Confusion matrix	Window close	Window open
Reality	Window close	50	0
	Window open	0	39

Table 5 : Confusion matrix for LR with undersampling.

For LR with oversampling (Table 6), the recall equals 97%.

		Predicted	
	Confusion matrix	Window close	Window open
Reality	Window close	10 474	0
	Window open	1	37

Table 6 : Confusion matrix for LR with oversampling.

Finally, for semi-supervised method, the AE model shows excellent detection ability with recall as 75%. The model learns the rule from window close case, so we can see from Table 7, the model did not classify any window opening case into window close case. By analysing the window opening prediction mistake, All 9 mistakes occurred during the winter period. The outside weather is more likely to have an impact on the indoor environment.

		Predicted	
	Confusion matrix	Window close	Window open
Reality	Window close	751	0
	Window open	9	27

Table 7 : Confusion matrix for AE.

All the results are compared in Table 8. In general, the supervised method performs better than the others. However, it has two drawbacks: (i) labelling all the data is time consuming and resource intensive and (ii) it is not applicable in the real world. For the unsupervised approach, the MG method and PCA show a good ability to detect activities. Due to the lack of labels and highly unbalanced data, these two methods show poor accuracy. Finally, the semi-supervised method has good precision and recall. As it requires little labelled data, it could be a satisfactory solution for the detection of occupant behaviour.

	Unsupervised		Supervised		Semi-supervised
	MG distribution	PCA	LR undersampling	LR oversampling	AE
Precision	0.32	0.07	1	1	0,75
Recall	0.88	0.81	1	0,97	1

Table 8 : Comparison of different methods.

5. CONCLUSION

In order to optimise the management strategies of smart buildings, different activity detection methods have been studied. Their advantages and disadvantages have been discussed. The results have been compared for the detection of windows openings of a virtual building: a lecture room simulated with EnergyPlus. The comparison shows that the supervised learning method has the best results, but it is not practically applicable. So, the semi-supervised approach can be a good alternative solution. In future work, a more realistic opening scenario will be generated, and the seasonality effect will be analysed. Then, real data collected in classrooms will be investigated with these methods. Ambient sensors, windows opening sensors and energy counters are collecting data since more than one month into two rooms. Data will be processed to detect the windows openings and the occupancy rate. A challenge in the detection process will be to distinguish if the outliers are due to windows events or to a change in the occupancy rate.

6. ACKNOWLEDGMENTS

This work was performed within the frame of the project BIoT (*Building Internet of Things*) of the RFI Wise program, of the French Region Pays de la Loire.

7. BIBLIOGRAPHY

Bishop, Chris M. 1994. “Novelty Detection and Neural Network Validation.” International Conference on Artificial Neural Networks. ICANN 1993: ICANN ’93 pp 789–794.

Bourlard, H., and Y. Kamp. 1988. “Auto-Association by Multilayer Perceptrons and Singular Value Decomposition.” *Biological Cybernetics* 59 (4): 291–94. https://doi.org/10.1007/BF00332918.

Chawla, N. V., K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. 2002. “SMOTE: Synthetic Minority Over-Sampling Technique.” *Journal of Artificial Intelligence Research* 16 (June): 321–57. https://doi.org/10.1613/jair.953.

Cox, D. R. 1958. “The Regression Analysis of Binary Sequences.” *Journal of the Royal Statistical Society. Series B (Methodological)* 20 (2): 215–42.

Danaei-mehr, Homay, Hüseyin Polat, and Aydın Çetin. 2016. *Resident Activity Recognition in Smart Homes by Using Artificial Neural Networks*. https://doi.org/10.1109/SGCF.2016.7492428.

Duda, Richard O. 2000. *Pattern Classification, Second Edition: 1*. John Wiley & Sons Inc.

Liu, Xu-Ying, Jianxin Wu, and Zhi-Hua Zhou. 2009. “Exploratory Undersampling for Class-Imbalance Learning.” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39 (2): 539–50. https://doi.org/10.1109/TSMCB.2008.2007853.

Paradiso, Francesca, Federica Paganelli, Antonio Luchetta, Dino Giuli, and Pino Castrogiovanni. 2013. “ANN-Based Appliance Recognition from Low-Frequency Energy Monitoring Data.” In *2013 IEEE 14th International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM)*, 1–6. https://doi.org/10.1109/WoWMoM.2013.6583496.

Stazi, Francesca, Federica Naspi, Marco D'Orazio. 2017. “Modelling window status in school classrooms. Results from a case study in Italy”. Building and Environment. 111. https://doi.org/10.1016/j.buildenv.2016.10.013.

Xilei Dai, Junjie Liu, Xin Zhang. 2020. “A review of studies applying machine learning models to predict occupancy and window-opening behaviours in smart buildings“. *Energy and Buildings*. 223. https://doi.org/10.1016/j.enbuild.2020.110159.