A Data-driven Meta-data Inference Framework for Building Automation Systems

Jingkun Gao
Civil and Environmental Engineering Department
Carnegie Mellon University
Pittsburgh, PA, USA
jingkun@cmu.edu

Joern Ploennigs
IBM Research
Smarter Cities Technology Centre
Dublin, Ireland
joern.ploennigs@ie.ibm.com

Mario Bergés
Civil and Environmental Engineering Department
Carnegie Mellon University
Pittsburgh, PA, USA
marioberges@cmu.edu

ABSTRACT

Building automation systems are believed to hold the key to significantly reducing the average energy consumption of our residential and commercial building stock, which in the U.S. is responsible for 41 % of the total annual energy use in 2014. As these systems become more widespread and inexpensive, the complexity and challenges associated with their installation, maintenance and upkeep will increase. One of the primary challenges is the generation and update of the meta-data associated with the sensors and control points distributed throughout the facility. Previous research has attempted to reduce the human input required to perform these activities, by leveraging different signal processing and statistical analysis approaches to infer the sensor types and locations from measurements and/or tags obtained through a BAS. However, because of the relatively small sample size, the feasibility of applying these type approaches on large buildings, as well as their generalizability, remain as unsolved questions. In this paper, we propose a meta-data inference framework to learn from BAS measurement data in a semi-automated way. Furthermore, we evaluate the framework on two large buildings instrumented with thousands sensors and show the feasibility of applying data-driven approaches in the real world. We present the results of our study and provide recommendations for future work in this area.

Categories and Subject Descriptors

I.5.4 [**Pattern Recognition**]: Applications—*Signal Processing*; H.4 [**Information Systems Applications**]: General; I.2.6 [**Artificial Intelligence**]: Learning—*Knowledge acquisition*

General Terms

Building automation systems

BuildSys'15, November 4–5, 2015, Seoul, South Korea..
© 2015 ACM. ISBN 978-1-4503-3981-0/15/11:.$15.00.
DOI:http://dx.doi.org/10.1145/2821650.2821670.

Keywords

buildings; meta-data inference; sensor type; data-driven approach; statistical feature

1. INTRODUCTION

Building Automation Systems (BAS) can help building managers and owners reduce the energy consumption of their facilities by 5 to 15 % according to some estimates [6]. Given that buildings account for approximately 41 % of the total annual energy budget of the U.S.[1], widespread adoption of BAS in the existing building stock can have a sizeable impact on the economy and the country's sustainability goals. This argument can, of course, be extended to other buildings and nations around the world.

Despite their purported benefits, in order to maintain their functionality and sustain their adoption, building automation systems bring about their own set of challenges to facility operators and owners. One such challenge is the generation and update of the meta-data associated with the many sensing, actuation, and control points managed by these systems. Meta-data here refers to any information associated with a device that helps to contextualize the measurements or control signals regularly being sent from/to the device, such as the location within the building, the physical phenomenon being sensed, etc. This meta-data is essential to enable effective use of the resources available to a BAS, yet in most situations it is either unavailable, unreliable, uninterpretable or outdated [7, 15].

Significant manual effort is currently required to generate and update meta-data in BAS. Such process is both time-consuming and error-prone. Meta-data is not only missing or mislabeled in the data collection system, but also highly unstructured and inconsistent as tags are added by different installers based on their own different naming schemes. The timely update of the meta-data when buildings evolve and change is also a challenge. Despite efforts to promote standard naming schemes such as Project Haystack [1], as well as efforts towards unified ontologies [13], no solution has been widely adopted. One reason may be that none of these naming schemes cover the wide variety of assets and information dimensions of meta-data described in [5]. Thus, recent research has focused on automating the process of acquiring, normalizing and extending the meta-data directly

[1] This value is for the year 2014, according to http://www.eia.gov/

Table 1: Example tags for BAS points in one building

WH19.TB7	WH19.TB8
WH19.EPM.3	WH19.ETD.2
STRT_HAMBAHU.AH10	BRG_BAKER160WING
HH5NHWT	UC20.SP3

from the information available in BAS, be it the tags [4, 3, 13, 15] or the measurements [10, 11].

In this paper, we present an approach to automatically associate sensor measurements with descriptive tags from a standard set (i. e., Haystack) by making use of the statistical distribution of individual sensor data. We cast the labeling problem as a supervised learning task and train different classification algorithms to learn the mapping from features of the measurements to Haystack tags. We evaluate our approach on two BAS datasets from large commercial buildings of over 100,000 sq. ft. containing over three years of combined historical time-series records.

2. PREVIOUS WORK

Previous research on automatically generating meta-data information mostly falls into two categories. The first category is interpreting meta-data from the text labels available in BAS. These labels are often obscure and only meaningful to local operators. Table 1 shows some labels from one of our datasets to illustrate this. We refer to this approach as Inference from Tags (IFT). Bhattacharya et al. [4] developed a synthesis technique to translate tags into useful metadata. The method can learn transformation rules by using a small number of examples provided by an expert. Schumann et al. [15] utilized linguistic and semantic techniques by first computing similarity values between BAS labels and then using semantic matching [14] to identify textual energy management systems inputs with minimal user involvement.

The second category analyzes available time-series from BAS to generate the meta-data information. Different signal processing and statistical techniques are used to extract useful features from the time-series which are then used, in conjunction with any existing labels, to infer the metadata for the rest of the sensing points. We refer to this approach as Inference from Measurements (IFM). Studies in this category mainly aim to find meta-data such as the spatial location of sensors (or actuators), the sensor type, the phenomena being measured, etc. Specially, a good portion of the literature focuses on sensor localization in buildings.

In [12] authors present an approach to automatically infer the spatial layout of rooms in a home, as well as the sensors in them. Ellis et al. [9] also proposed an approach to infer room connectivity by using light and motion sensor data. Hong et al. [10] applied empirical mode decomposition on data from 15 environmental sensors across 5 rooms to find the sensors which are in the same room by analyzing correlation coefficients of intrinsic mode functions. Koc et al. [11] explored how to infer the relative locations of temperature sensors with respect to each other from sensor measurements, by using linear correlation and a statistical dependency measure. Rather than determining the relative locations, Akinci et al. [2] studied the possibility of using sensor data related to Heating Ventilation Air-Conditioning (HVAC) equipment to identify the exact room in which the sensor is located, by combining sensor measurements and building characteristics. Apart from location information, Calbimonte et al. [8] derived semantic meta-data of the sensor types using piecewise linear representation of raw time-series measurements, with the assumption that data from same type of sensor will produce similar patterns. Xiao and Cheng [17] proposed a framework for mining BAS datasets using data mining techniques including cluster analysis and association rules, though domain knowledge is still needed for the knowledge discovery.

All proposed methods in this category are basically trying to infer meta-data from measurements using a model trained on a small portion of data. In that sense, our approach falls under the IFM category, and is not different from the previous work. However, the work presented in this paper distinguishes itself in two ways: (a) simplicity: we posit that the distribution of the measurements, as captured by some descriptive statistics, contains sufficient information to allow a classifier to effectively discriminate tags; (b) scope of evaluation: our tests are carried out in datasets that are longer and more varied than those used in previous work. In the next section we introduce our method in more detail.

3. META-DATA INFERENCE FRAMEWORK

Figure 1 shows a block diagram of the meta-data inference framework used in this paper. It consists of five components, including data pre-processing, feature extraction, model training, meta-data inference and meta-data integration. The data pre-processing step performs some basic transformations to the raw data such as resampling, interpolation and outlier removal. Some of these pre-processing steps are performed manually or under human supervision and are not strictly part of the framework. Details are provided in Section 4.

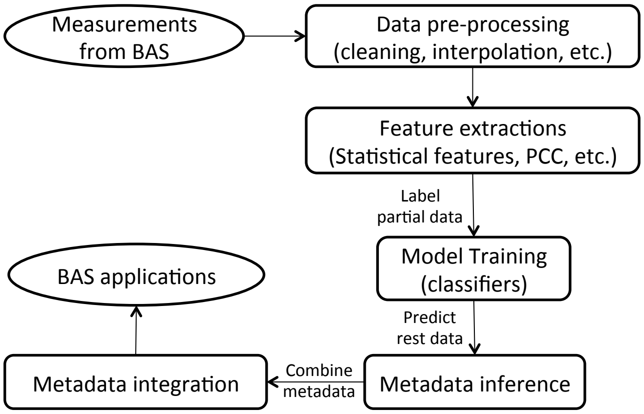

Measurements from BAS
Data pre-processing (cleaning, interpolation, etc.)
Feature extractions (Statistical features, PCC, etc.)
Label partial data
Model Training (classifiers)
Predict rest data
Metadata inference
Combine metadata
Metadata integration
BAS applications

Figure 1: Meta-data inference framework

After pre-processing, each one of the time series of a sensor, actuator, or control measurement is referenced as a data point $i \in \{1, \ldots, N\}$ that is considered to be a random variable $x_i \in \mathbb{R}_i^T$, where T is the number of samples in the time period of analysis. Also, each x_i has an associated label y_i that describes it (e. g., 'Return Air Temperature'). In general, we can say that $y_i \in \Lambda$ and that Λ is the set of all possible labels, i. e. the meta-data information that is of our interest.

In the feature extraction step, summary statistics of the data for each label y_i are extracted to describe the distribution $P(x_i|y_i)$. Although we began our experiments with a large set of candidate statistical and frequency-based features describing both the marginal distribution $P(x_i|y_i)$ and the joint distributions $P(x_1, \ldots, x_N)$, the descriptive statics of the marginal distributions fared better. Thus, the feature extraction step simply computes the mean, median, mode, quantiles and deciles for the measurements belonging to each label y_i in order to capture the salient features of this distribution.

Model training, the third step in the diagram, involves using labeled data (i. e., tuples $< x_i, y_i >$) to learn a mapping $f : X \to Y$. In some cases, it may be more convenient learn binary mappings $f_j : X \to \{0, 1\}$ for each of $j \in \Lambda$, i. e. a binary classifier for each label. In the meta-data inference step, the trained models (from here on referred to as classifiers) are used to predict the labels for the remaining portion of the dataset. The last step is to integrate all the meta-data and make it available for the different BAS applications.

In this paper, we are primarily interested in the sensor type (e. g., '`Return Air Temperature`') out of the different categories of meta-data [5]. The underlying assumption is that the measurements from the same type of sensors should have similar statistical distributions. However, we believe this assumption applies to other pieces of meta-data information such as the location of sensors (though, perhaps only after conditioning on the sensor type).

It is also worth noting that the individual elements of the set of labels we are concerned with (Λ) can be thought of as composite labels. In other words, one can consider labels such as '`Return Air Temperature`' to be a composition of many individual tags (in this case '`return`', '`air`' and '`temperature`'). Given this, a more general description of the framework would be to consider that Λ is the superset of all possible individual tags found in some schema, such as the one provided by Project Haystack[2]. Relying on such a schema also standardizes the mappings that are learned and could make the approach transferable from one building to another. In general, we will make a clear distinction between models trained to predict individual tags, and those trained to predict composites of such tags (which we will sometimes refer to as composite labels).

Following this logic, we convert all BAS tags found in the datasets described in the next section into Haystack standard tags. Some examples can be seen in Table 2. To complete this conversion, sometimes we needed to rely on significant human expertise as the information contained in several BAS labels was incomplete and need to be extended to map it to a proper Haystack label. For example, '`Airflow Request`' was extended to '`vav air flow cmd`' referencing the air flow control command (CMD) of the variable air volume (VAV) asset. Similarly, the BMS label '`Cool Request`' was extended to '`ahu cool cmd`' that is referring to the air handling unit (AHU) cooling control command.

4. TESTBED AND DATASET

We now describe the two buildings that were used as testbeds for evaluating the inference framework. Both are large facilities (over 150,000 sq.ft.), though they are located in different climate zones and have different uses. To facilitate reading, we have named the datasets GHC and DUB and will continue to use those acronyms throughout the rest of the paper.

Table 2: Some examples of BAS tags and their corresponding Haystack tags

BAS	Haystack
Airflow	vav air flow sensor
Airflow Request	vav air flow cmd
Airflow Setpoint	vav air flow sp
CHW Valve VDC	ahu chilled water valve vfd
Cool Request	ahu cool cmd
Damper Position	vav damper
Discharge Air Temp Setpoint	vav discharge temp sp
Discharge Temp	vav discharge temp sensor

4.1 Gates and Hillman Centers (GHC)

The Gates Center for Computer Science and Hillman Center for Future Generation Technologies is a 217,000 sq. ft. facility with nine floors, built in 2009 at Carnegie Mellon University, Pittsburgh, Pennsylvania, USA. It contains approximately 310 offices, 11 conference rooms, 32 laboratories and a number of other spaces. We have access to historical records from the BAS installed in this facility, for the months of December 2014 and June 2015. During these periods, the BAS recorded 7,310 time series at a temporal resolution of roughly 16 minutes. The sensors range from HVAC system instruments (e. g., air flow sensor, damper position) to environmental sensors (e. g., CO2 level, humidity), and electrical power sub-meters. Each time series is associated with a tag specifying its meta-data.

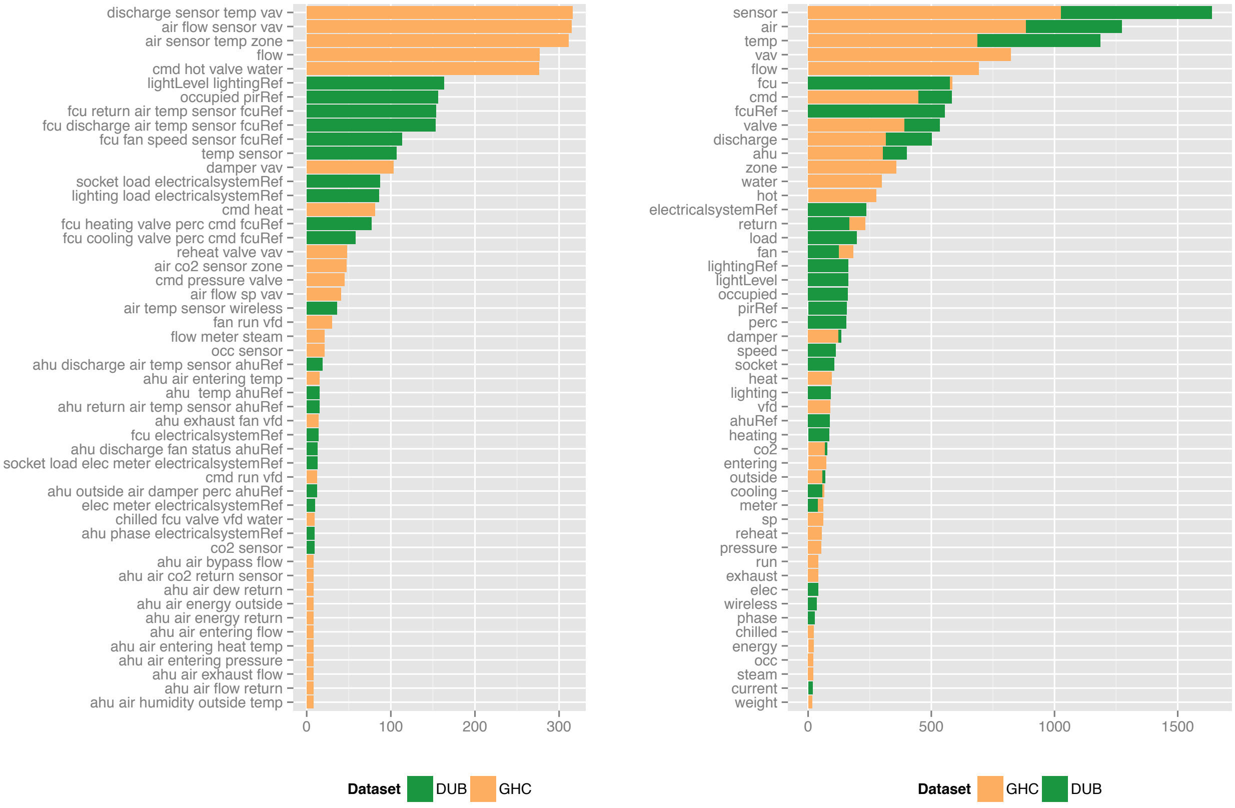
The pre-processing step adopts a conservative strategy to remove broken sensors that do not update their values. They are characterized in the dataset by a steady output or missing values. Therefore, we remove sensors with more than 95 % of the measurements being the same for the duration of study, as well as certain time series with less than 1000 samples. The remaining 2,354 time series were interpolated to the same sampling period of 15 minutes. This relatively conservative strategy could be relaxed in the future studies to incorporate more disturbances and develop more robust methods.

A total of 53 different Haystack tags were used, which occur in 94 unique composite combinations. We also remove all composite labels with a frequency lower than 5 to ensure that at least abundant examples (more than 5 for each label) were present in the dataset to train and test the classifiers. This resulted in a significantly smaller subset of only 56 common composite labels composed of 43 individual tags used by all 2,256 data points. Figure 2(a) and Figure 2(b) show the histogram for the composite and individual tags with more than 50 counts, respectively. It is easy to see that the occurrence of unique tags and composite labels is Pareto distributed, and that is also true for the second dataset described in the next subsection. This explains why a majority (96 %) of the data points (2256/2354) is actually using a smaller portion (81 %) of the tags (43/53) and 60 % of the tag combinations (56/94).

4.2 SCTC Living Laboratory (DUB)

The IBM Research SCTC Living Laboratory in Dublin is

[2] http://project-haystack.org/



(a) Composite labels

(b) Individual tags

Figure 2: Histogram of 50 most frequent tags in both datasets

a modern 161,000 sq. ft. office building. The building was converted to office usage in several steps over the last 3 years from an old assembly line. It is highly equipped with modern building automation technology to provide a rich data source for analytics research. It provides about 3,000 data points ranging from HVAC equipment (e. g., boilers, chillers), indoor condition sensors (e. g., temperature, brightness, CO2, humidity), occupancy sensors, and electric meters. For some sensors, the historical time series record goes back approximately 3 years to 2012 with varying sampling frequency. The monitoring system collects data from 13 different subsystems communicating under different bus protocols such as BACnet, LON, Dali, Modbus, ZigBee and proprietary ones. The initial labeling of the data points is very diverse due to the variety of technologies and the different contractors involved over the building life cycle.

1,678 time series were selected for which more than 60 % of historical values were available. These time series were labeled against an extended Haystack label set, just as we did with the GHC dataset. The extended set uses 109 different tags, which combine to 176 unique composite labels. By removing all composite labels with an occurrence count lower than 5, we again drastically reduce the variability of tags used in the dataset to only 37 tags (34 %) in 28 unique combinations (16 %) that apply to 1372 time series (82 %). Figure 2 confirms the Pareto distribution for the DUB dataset and may generate optimism that the classification problem can be solved in the experiments conducted in the next section.

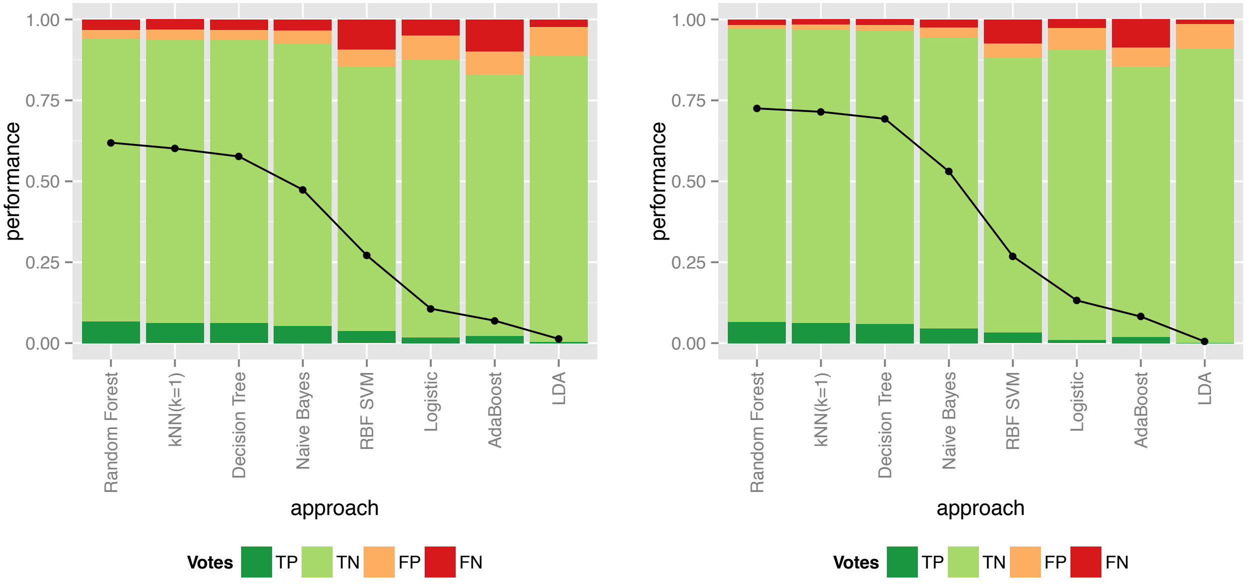
5. EXPERIMENTS

We now leverage the GHC and DUB datasets to evaluate the performance of the framework described in Section 3. We carry out three types of experiments: one where the labels y_i belong to the set of all composite labels found in each dataset; one where y_i are individual tags contained in those composites, and where we train binary classifiers for each one of them; and finally one where we combine the output of all these binary classifiers in an attempt to recreate the composite label. Each one of these experiments, along with the results, are described in more detail below.

5.1 Classification of composite labels

In the last section we observed a Pareto distribution in the usage of tags and composite labels in the datasets. This is an essential pre-requisite for our approach, as it ensures that the classification problem needs to manage only a small degree of variability to cover a large amount of the data.

Based on the optimism caused by this finding, we investigate first if it is possible to identify the full composite label of a time series out of its statistical properties. This is a multi-class classification problem, as the classifier has to identify the correct composite label (e.g., ‘temp sensor’) out



(a) GHC dataset

(b) DUB dataset

Figure 3: Average classification performance of different classification approaches

of all existing composite combinations (‘`temp sensor`’, ‘`occ sensor`’, ‘`co2 sensor`’, etc.).

Figure 3 shows the classification performance of different off-the-shelf classification algorithms, namely Random Forest, k-Nearest Neighbors (kNN), Decision Trees, Gaussian Naïve Bayes, Support Vector Machines with Radial Basis Function kernels (RBF SVM), Logistic Regression, AdaBoost and Linear Discriminant Analysis (LDA). Each classifier is trained for each dataset on 20 % of the whole set and then tested against a disjunct set of the remaining 80 %. The training set is selected in a stratified way where 20 % of the examples belonging to each label are retained. We execute 30 experiments with different randomly selected training and testing sets. Figure 3 shows the weighted mean true positive (TP), true negative (TN), false positive (FP), and false negative (FN) rates for predicting the correct composite label using the 8 different classifiers. We are using this set of performance metrics to ensure comparability with the multi-label classification problem discussed in the next section.

Though the FP and FN rates seem relatively small, the results may seem overly optimistic because of the high TN rate. Given this, we also computed the F1 score defined as $2\,\mathrm{TP}/(2\,\mathrm{TP}+\mathrm{FP}+\mathrm{FN})$, as shown in the dark line plot found in the figure. It is also worth noting that the classification algorithms are sorted by their F1 score, where the best performing approaches have a higher F1 score. The resulting ranking of classification approaches is identical for both datasets, as shown in Figure 3(a) and Figure 3(b). The best performing approaches are Random Forest, k-Nearest Neighbour, and Decision Trees. They are characterized by a high true positive rate and low false negative rate. Naïve Bayes and RBF SVM perform less well and AdaBoost, Logistic Regression and LDA show the worst performance.

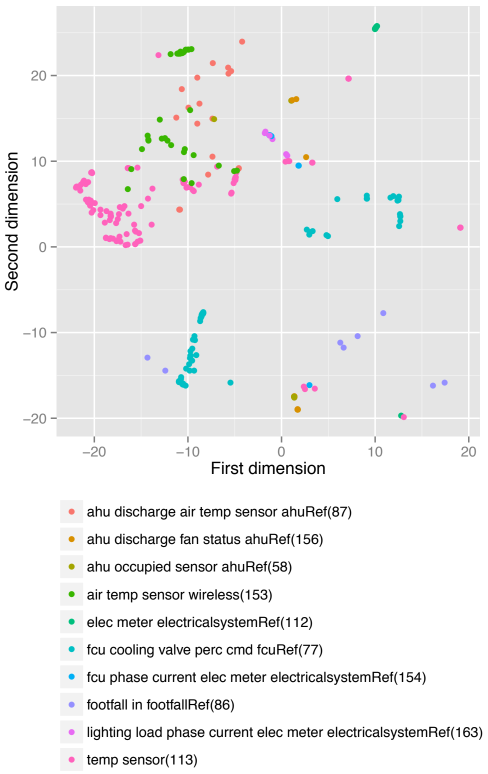


Figure 4: Scatter plot of the 2D t-SNE features for the DUB dataset

Particularly, the bad performance of AdaBoost is surprising as, like Random Forest, it is an ensemble method that uses Decision Trees as the base estimator. However, both Random Forest and Decision Trees significantly outperform AdaBoost. The reason lies within the statistical features of the time series that are used. They are not well differentiated by hyperplanes, but instead, occur often in similar clusters in different composite labels. For example, let us consider two fan coil units (FCUs) supplying different rooms. One is operated at a setpoint of 21 °C and the other at 18 °C. Both FCUs have a sensor labeled '`supply air temperature`' that will measure temperature values in quite different ranges. If the '`return air temperature`' of the first FCU is about 3 °C lower, then it will be in a similar range with the '`supply air temperature`' of the second FCU. To further understand this, we look at Figure 4, the scatter plot of the feature space of DUB dataset reduced to 2 dimensions by t-Distributed Stochatic Neighbor Embedding (t-SNE) [16]. The figure shows the 10 most frequent composite labels where the number beside the label represents the count. Some labels are well discriminated such as '`fcu cooling valve perc cmd fcrRef`', but others are clustered together such as '`ahu occupied sensor`' which overlaps with many other categories. These overlaps also explain why classification approaches with a less flexible decision boundary like Naïve Bayes, Logistic Regression, or LDA cannot separate the hyperspace well. The k-nearest Neighbor classifier and Decision Trees are better able to adapt to the overlapping cluster. The bagging approach used by Random Forest divides the training set into random subsets, which further increases its chance of finding good separations. The boosting algorithm used by AdaBoost is weighting the training samples and overfits to specific properties in the training set, which results in a high false negative rate in both datasets.

Figure 5 shows the classification performance for the specific composite labels in the datasets, using just the Random Forest approach. The F1 score decreases nearly linearly across the composite labels in both datasets, such that it is higher than 0.5 for about 50 % of the labels. The classification performance reaches very high scores for '`ahu humidity outside temp`' and '`cmd hot valve water`' or '`air sensor temp zone`' in the GHC dataset. The '`temp sensor`' has an acceptable F1 score in the DUB dataset. The best performing labels are '`CO2 sensor`' and '`occupied`' (PIR sensor) as well as '`elec meter`'. This is also explainable by the well distinguished data ranges of these sensor types in the DUB dataset. Figure 4 shows '`temp sensor`' is characterized by well defined clusters in the 2D embedded space.

An important question for the identification of sensor metadata is the required size of the training sample. Figure 6 shows how the classification performance changes with training set sizes ranging from 20 % to 90 % of each dataset. Interestingly, the training set size has virtually no influence on the classification performance of the GHC data set. The overall performance for this dataset remains at a F1 score of 0.6. This indicates that the features are overlapping too much in the dataset and cannot be separated, regardless of the training set size. The classification performance of the DUB dataset slightly increases with the training set size from an F1 score of 0.75 to 0.8.

We can conclude that the classification performance for the composite label problem is not optimal, and is far from becoming practical. Even the best performing classifiers (i. e., Random Forest, k-NN and Decision Trees) reach a mean F1 score of 0.6 for the GHC and 0.7 for the DUB dataset.

This conclusion is valid not only for the features we selected for this study, but, actually also for other features that were investigated such as dominant frequencies, regression parameters, and information density features. However, we also showed that the classification performance is not equally distributed among the labels. Thus, some labels are usually well separated from the rest in feature space. Consequently, in the next sub-section we train individual classifiers on each of the labels in an attempt to reduce the complexity of the classification task.

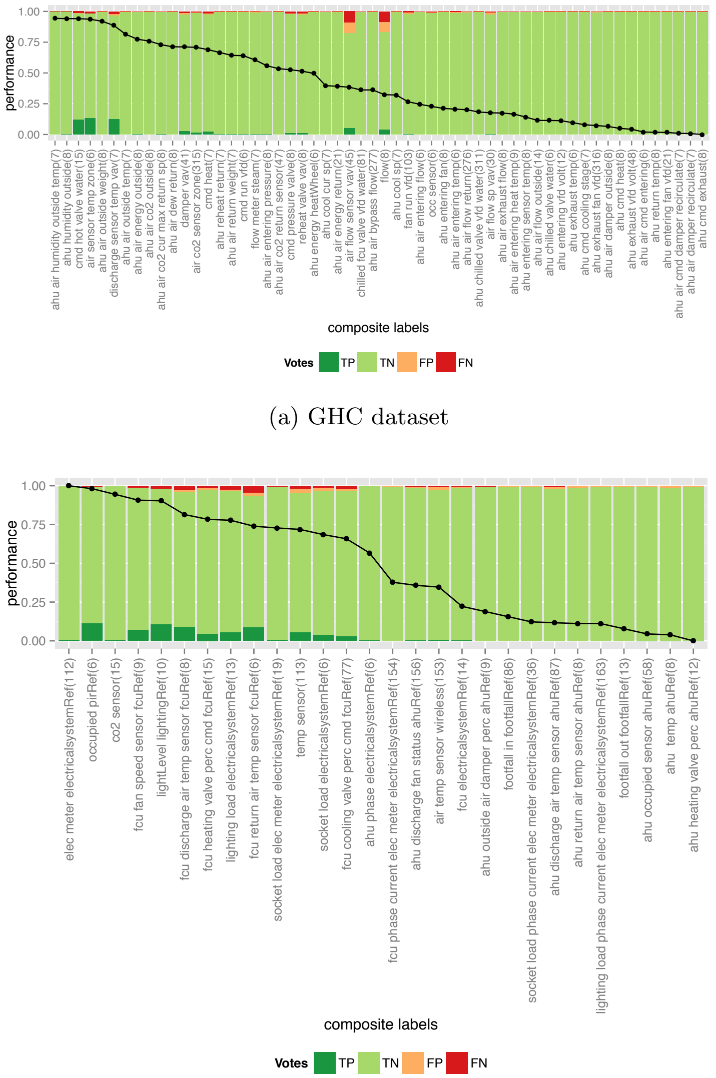
5.2 Classification of individual tags

Here we investigate the performance of binary classifiers trained on individual tags. Multiple individual tags ('`temp`' or '`sensor`') can be assigned to each time series. Each classifier is trained with 20 % of the data and tested with the remaining 80 %. The experiments were conducted 30 times and the average weighted performance for each tag is reported. We used the same eight classifiers as in the previous section, and the best results were obtained with Random Forest, kNN and AdaBoost. Figure 7 shows the TP, TN, FP, FN and F1 score for each individual tag and for both dataset with the kNN classifier, which yields the highest average F1 score. As was the case in earlier figures, the tags in the horizontal axis are also sorted by their F1 scores.

For the GHC dataset, we can see that the 6 tags with F1 scores higher than 90 % are '`cool`', '`hot`', '`water`', '`heatWheel`', '`temp`', and '`valve`'. For all these individual tags we can see they are all very sensitive to temperatures, which makes them stand out over other individual tags. For the remaining tags in the high-performer set ('`max`', '`cmd`', etc.), they all have some unique features to be identified and are also present in only a few composite labels. Unlike the first few ones, three tags showing significant FP and FN values are '`sensor`', '`air`' and '`vav`'. '`sensor`' refers to the type of certain monitoring points, which is a tag widely used by many points in BAS. The same holds true for tag '`air`' and '`vav`'. Thus, in order to achieve high identifyability, the tag itself should not be too broad (unless it is associated with some characteristic patterns), since this will limit its discrimination. Another reason is that many BAS tags belonging to sensors are not labeled as '`sensor`', which will introduce false positives.

For the DUB dataset, 30 % of the total individual tags show good performance with an F1 score of over 80 %. The overall better performance from the DUB dataset may be related to the fact that data from this set are collected for approximately 3 years while in the GHC dataset only one-month-long periods are analyzed. The future study will incorporate a longer time series of data from GHC to conduct analysis. The two tags which give highest FP and FN values are '`fcuRef`' and '`fcu`'. '`fcu`' refers to fan coil unit (FCU) consisting of heating or cooling coil and fan, which has its own thermostat to control the temperature. Any BAS points inside FCU equipment or the rooms that a FCU is supplying air to could be labeled as '`fcu`', which will inevitably have diverse distributions and become hard to recognize.

Exploration on individual labels also shed light into the functionality of different sensors, i. e., sensors can have the



(a) GHC dataset

(b) DUB dataset

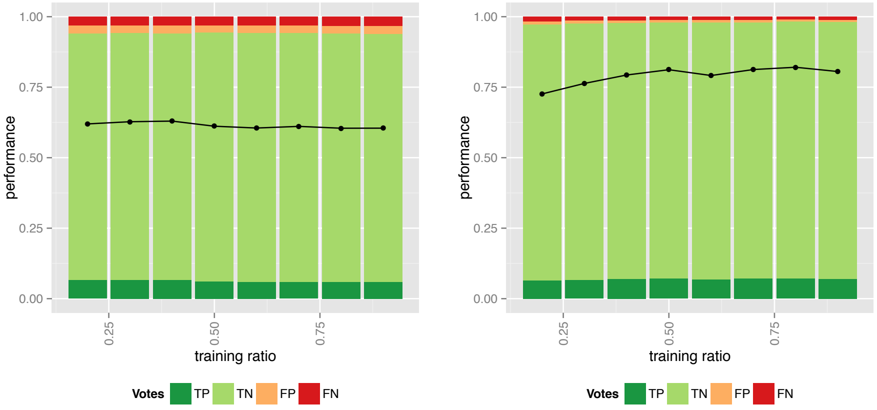
Figure 5: Classification performance for different composite labels using random forest

same individual tag despite being of different types. One example is that all the sensors functioning together with air handling units do not have to be of the same type or even in the same location. The de-coupling of composite labels helps to understand more information associated with the buildings and to infer more meta-data from measurements. In the next subsection, we will show how to combine all the binary classifiers together to predict the composite labels.

5.3 Combination of individual labels

Here we are interested in merging the results of all binary classifiers to create a composite label. A simple but naïve way of doing this is to assume that the composite label is the union of the label predicted by each binary classifier. For instance, if the only tags that were predicted were '`outside`', '`air`' and '`humidity`' (i. e., those were the only classifiers that gave a positive result), then the composite tag would be '`outside air humidity`'. However, this naïve approach without constraints allows for many more composite labels than what may be present in the training dataset. Thus, a more promising idea is to constrain the composite labels to be only the set contained in the ground truth (or in some other reference set). We can do this by calculating the Euclidean distance between the vector of the probabilities of each of the tags and the ground truth binary vector of every composite label in the reference set. A final voting decision is made by assigning the new example to the composite label with the smallest Euclidean distance.

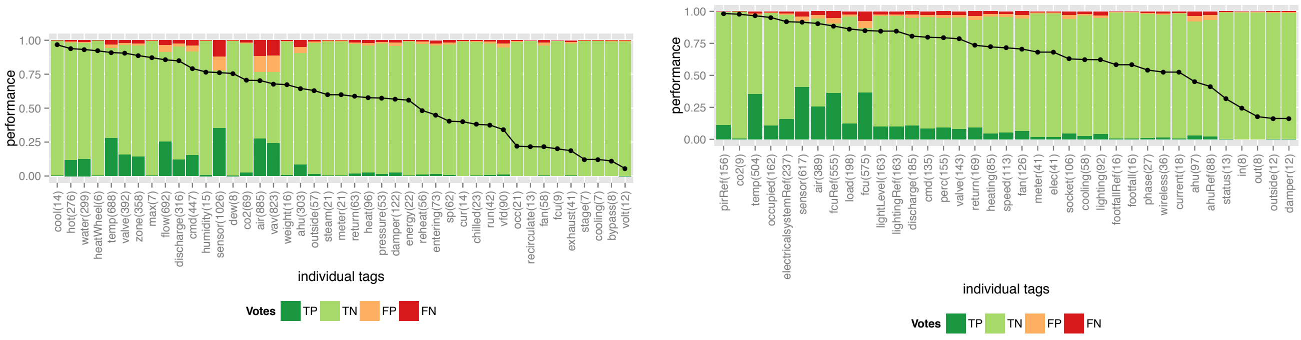
Figure 8 shows the performance of combining binary classifiers to predict composite labels using Random Forest. The top two subplots depict results from the GHC dataset. On the right we show the unconstrained case and on the right the constrained one. The bottom two show the performance of the DUB dataset. From the figure we can see a modest improvement in the F1 score by adding the constraint to reduce the space of possible composite labels.



(a) GHC dataset

(b) DUB dataset

Figure 6: Composite labels training ratio using random forest



(a) GHC dataset

(b) DUB dataset

Figure 7: Classification performance of different individual tags using kNN (k=1)

5.4 Training and testing on different datasets

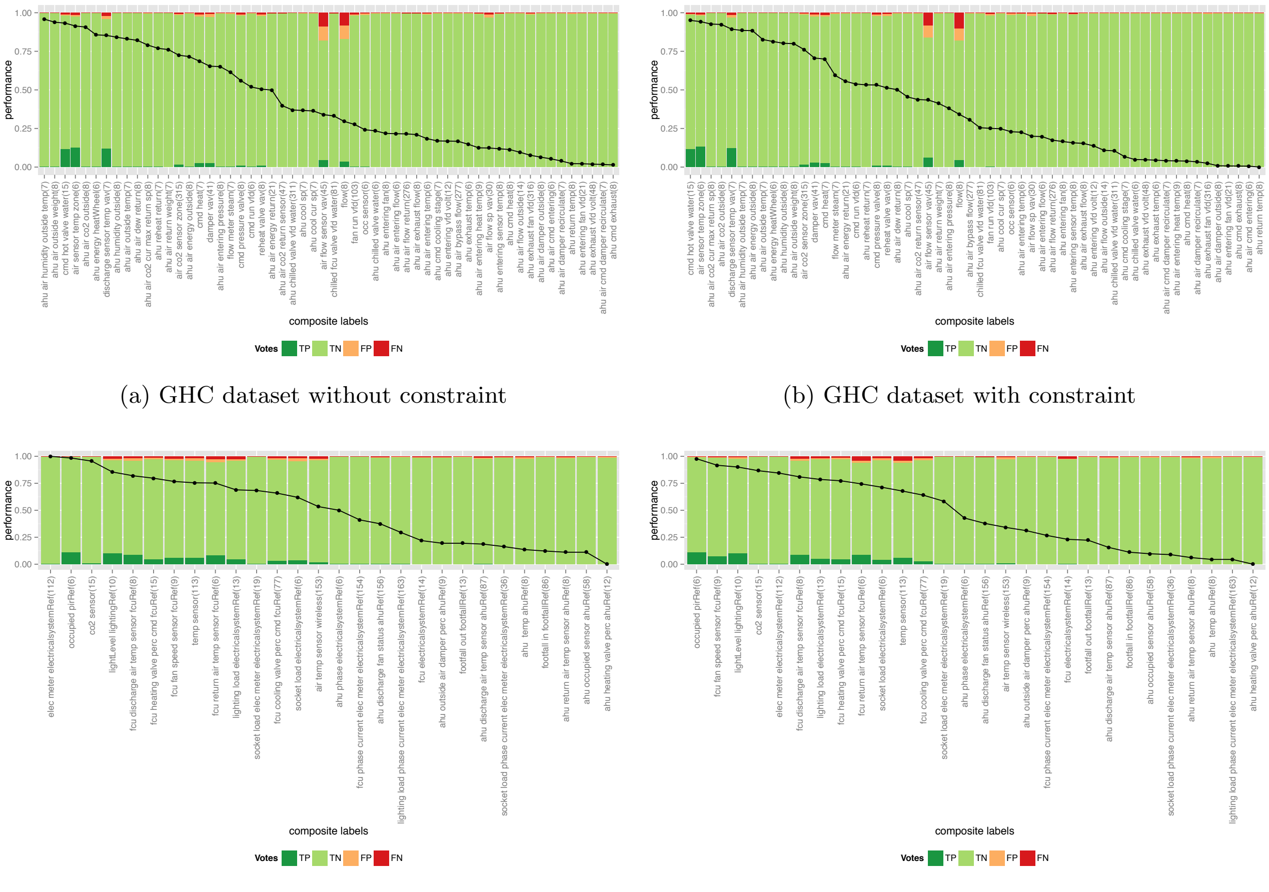
After discussing how individual labels can be used to infer composite labels, the next question to be answered is: is it possible to train on one dataset and test on the other one? An affirmative answer can have important implications on the scalability and feasibility of the approach for real-world applications.

One of the most challenging issues of such problem, however, is to make sure labels/tags are from the same name space and used with the same meaning. To start simple, we first train and test on the GHC dataset but in different periods (December, 2014 and June, 2015), to represent data from two different seasons. Even though they are from the same building, due to the missing data in different months, their tags do not match exactly. Thus the intersection of the individual tags from each month is used. Moreover, this experiment will also allows to test the robustness of the approach to the seasonality that the statistical distributions of some time-series may have. Table 3 shows the F1 score for Random Forest. There are many other tags including '`cool`', '`hot`', '`pressure`' with a very low 0 % F1 score in both cases. Those are not listed on the table. The fact that the performance varies significantly for different tags is expected since the distribution of the measurement is expected to change seasonally, especially for temperature sensitive measurements. On the other hand, some individual tags, such as '`outside`', '`air`', '`fcu`' and '`cur`' seem to be less sensitive to the seasonal change.

Another test case is to train on the GHC dataset and test on DUB dataset, and the other way around. However, this brings the challenging part of the labeling problem. Even though both sets tried to use HayStack convention to transform BAS tags into a common name space, there are still a lot of debates of how to name certain specific tags. Besides, the duration of the collected data in two buildings is also different, which further complicates the problem. Nonetheless, we ran a set of experiments to test this idea and the preliminary results showed a very low performance for almost all individual tags except for a few: '`temp`' at 54 % and '`valve`' at 78 %. More explorations on training and testing across datasets will be performed as part of future work.

6. CONCLUSIONS AND FUTURE WORK

We presented a simple data-driven approach to infer the meta-data associated with BAS sensors in a semi-automated way. We formulated the problem as a supervised classifica-



(a) GHC dataset without constraint

(b) GHC dataset with constraint

(c) DUB dataset without constraint

(d) DUB dataset with constraint

Figure 8: Classification performance of composite lables by making use of individual tags

Table 3: Training and testing in different periods for individual tags

	outside	air	fcu	cur
Train on Dec:	79.66%	70.73%	67.89%	64,91%
Train on Jun:	82.51%	72.16%	66.66%	72.54%

tion problem and utilized statistical features describing the distribution of the measurements to represent each sensor tag. This approach is computationally efficient, and produces acceptable results (more than 50 % F1 score for half composite labels and more than 60 % F1 score for half individual tags) with a small portion of training examples (20 % of the dataset). Accuracy ($\frac{TN+TP}{TN+TP+FN+FP}$) is not reported here, since the high TN values are biasing the result producing an average value of up approximately 90 % for most cases.

To investigate the applicability of our approach, we tested on two different buildings in different climates and serving different functions. The results show that when training/testing in the same facility, the approach can predict individual tags with an average accuracy of 95 %, though by inspecting the F1 score we find that this result is highly dependent on the specific meta-data tag that is being inferred. Inferring composite labels produced less favorable results. The best results in this case were achieved when training/testing for those tags that had more unique patterns, and this produced a weighted average F1 score of 75 %. It is noted that the results depend on the features selected and future work will investigate into additional features with better performance.

Further, we investigated into the sensibility of the approach to seasonal effects by first training and testing the model at specific months of the year; and then applying a test set from a different time period (or season), thus verifying the stability of the learned model. Our results indicate that, as expected, the models need to be updated as the distribution of the sensor measurements changes with time. However, rather surprisingly, some tags such as '`outside`' and '`air`' show seasonally insensitive patterns in different periods.

Finally, we conducted preliminary experiments with training and testing across buildings, given that a favorable result in this case would mean that the approach generalizes well and could be applied to a new building without training a subset of a-priori labeled data. In this case, the results were less conclusive and this is where we believe significant amount of effort needs to be put forth by the community to make these data-driven approaches practical.

Though our approach is supervised and requires a subset of the data to be fully labeled, we believe the results seem promising enough to encourage additional work in this area given that, while not fully automating the process, a significant amount of the manual labor required for labeling BAS points in buildings can be reduced by implementing this strategy.

7. ACKNOWLEDGMENT

The authors would like to acknowledge China Scholarship Council (CSC) for the funding that supported the research presented in this paper.

8. REFERENCES

[1] Project haystack. http://project-haystack.org/. Accessed: 2015-07-23.

[2] B. Akinci, M. Berges, and A. G. Rivera. Exploratory Study Towards Streamlining the Identification of Sensor Locations Within a Facility. In *Computing in Civil and Building Engineering*, pages 1820–1827. ASCE.

[3] A. Bhattacharya, D. Culler, D. Hong, K. Whitehouse, and J. Ortiz. Writing scalable building efficiency applications using normalized metadata. In *BuildSys*, pages 196–197. ACM Press, Nov. 2014.

[4] A. Bhattacharya, D. E. Culler, J. Ortiz, D. Hong, and K. Whitehouse. Enabling Portable Building Applications through Automated Metadata Transformation. Technical Report UCB/EECS-2014-159, EECS Department, University of California, Berkeley, Aug. 2014.

[5] A. Bhattacharya, J. Ploennigs, and D. Culler. Analyzing metadata schemas for buildings: The good, the bad, and the ugly. In *BuildSys*. ACM Press, Nov. 2015.

[6] M. R. Brambley, P. Haves, S. C. McDonald, P. A. Torcellini, D. Hansen, D. Holmberg, and K. Roth. *Advanced sensors and controls for building applications: Market assessment and potential R & D pathways.* Pacific Northwest National Laboratory Washington, DC, USA, 2005.

[7] J. F. Butler and R. Veelenturf. Point naming standards. *ASHRAE Journal*, 52:B16, 2010.

[8] J. Calbimonte, O. Corcho, Z. Yan, H. Jeung, and K. Aberer. Deriving semantic sensor metadata from raw measurements, Oct. 2012.

[9] C. Ellis, J. Scott, I. Constandache, and M. Hazas. Creating a room connectivity graph of a building from per-room sensor units. In *BuildSys*, page 177. ACM Press, Nov. 2012.

[10] D. Hong, J. Ortiz, K. Whitehouse, and D. Culler. Towards Automatic Spatial Verification of Sensor Placement in Buildings. *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings - BuildSys'13*, pages 1–8, 2013.

[11] M. Koc, B. Akinci, and M. Bergés. Comparison of linear correlation and a statistical dependency measure for inferring spatial relation of temperature sensors in buildings. In *BuildSys*, pages 152–155. ACM Press, Nov. 2014.

[12] J. Lu and K. Whitehouse. Smart Blueprints: Automatically Generated Maps of Homes and the Devices Within Them. In J. Kay, P. Lukowicz, H. Tokuda, P. Olivier, and A. Krüger, editors, *Pervasive Computing SE - 9*, volume 7319 of *Lecture Notes in Computer Science*, pages 125–142. Springer Berlin Heidelberg, 2012.

[13] J. Ploennigs, B. Hensel, H. Dibowski, and K. Kabitzsch. BASont - A modular, adaptive building automation system ontology. In *IECON - 38th An. Conf. on IEEE Ind. Electr. Soc.*, pages 4827–4833, Oct. 2012.

[14] A. Schumann and F. Lécué. Minimizing user involvement for accurate ontology matching problems. In *29th AAAI Conf. on Artificial Intelligence*, pages 1576–1582. AAAI Press, 2015.

[15] A. Schumann, J. Ploennigs, and B. Gorman. Towards automating the deployment of energy saving approaches in buildings. In *BuildSys*, pages 164–167. ACM Press, Nov. 2014.

[16] L. Van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

[17] F. Xiao and C. Fan. Data mining in building automation system for improving building operational performance. *Energy and Buildings*, 75:109–118, June 2014.