Energy Disaggregation for Real-Time Building Flexibility Detection

Elena Mocanu, Phuong H. Nguyen, Madeleine Gibescu
Department of Electrical Engineering, Eindhoven University of Technology
5600 MB Eindhoven, The Netherlands
Email: {e.mocanu, p.nguyen.hong, m.gibescu}@tue.nl

Abstract—**Energy is a limited resource which has to be managed wisely, taking into account both supply-demand matching and capacity constraints in the distribution grid. One aspect of the smart energy management at the building level is given by the problem of real-time detection of flexible demand available. In this paper we propose the use of energy disaggregation techniques to perform this task. Firstly, we investigate the use of existing classification methods to perform energy disaggregation. A comparison is performed between four classifiers, namely Naive Bayes, k-Nearest Neighbors, Support Vector Machine and AdaBoost. Secondly, we propose the use of Restricted Boltzmann Machine to automatically perform feature extraction. The extracted features are then used as inputs to the four classifiers and consequently shown to improve their accuracy. The efficiency of our approach is demonstrated on a real database consisting of detailed appliance-level measurements with high temporal resolution, which has been used for energy disaggregation in previous studies, namely the REDD. The results show robustness and good generalization capabilities to newly presented buildings with at least 96% accuracy.**

I. Introduction

Energy is a limited resource which faces additional challenges due to recent efficiency and de-carbonization goals worldwide. An important component of the ongoing process is the improvement in the energy management systems in residential and commercial buildings, which account for $30-40\%$ of the total energy demand in the developed world [1]. Buildings are complex systems composed by a different number of devices and appliances, such as refrigerators, microwaves, cooking stoves, washing machines etc. However, there are also a number of sub-systems, e.g. electric heating, lighting. Even there are many influencing factors in building energy consumption, some patterns can be clearly identified and used further to improve demand side management systems and demand response (DR) programs [2]. Identifying and aggregating the flexibility resource at the community level can decrease the end-user energy bill. Concomitantly, as a long-term benefit, flexibility can lead also to emission reductions, and lower investments in transmission and distribution grid infrastructure. Therefore, the role of end-users and their available flexibility is becoming increasingly important in the Smart Grid context.

One possible way to detect building flexibility in real-time is by performing energy disaggregation. Disaggregation refers to the extraction of appliance level energy signals from an aggregate, or the whole-building, energy consumption signal. Often only this aggregated signal is made available via the smart meter infrastructure to the grid operator, due to privacy concerns of the end user. This new approach should open new paths towards better planning and operation of the smart grid, helping the transition of end-users from a passive to an active role. In addition, informing the end-user in real-time, or near real-time, about how much energy is used by each appliance can be a first step in voluntarily decreasing the overall energy consumption.

Introduced by W. Hart [3] in the early 1980s, the Non-Intrusive Load Monitoring (NILM) problem has nowadays several solutions for residential buildings. Traditional approaches for the energy disaggregation problem (or NILM problem) start by investigating if the device is turned on/off [4], and followed by many steady-state methods [5] and transient-state methods [5] aiming to identify more complex appliance patterns. In the same time, advance building energy managements systems are looking beyond quantification of the energy consumption by including fusion information such as, the acoustic sensors to identify the operational state of the appliances [6], the motion sensors, the frequency of the appliance used [7], as well as time and appliance usage duration [7], [8]. A more comprehensive discussion about these can be found in recent reviews, such as [9]–[11]. Moreover, new data analytics challenges arise in the context of an increasing number of smart meters, and consequently, a big volume of data, which highlights the need of more complex methods to analyze and take benefit of the fusion information [12]. More recent researches have explored a wide range of different machine learnings methods, using both supervised and unsupervised learning, such us sparse coding [8], clustering [13], [14] or different graphical models (e.g. Factorial Hidden Markov models (FHMM) [7], Factorial Hidden Semi-Markov Model (FHSMM) [7], Conditional FHMM [7], Conditional Factorial Hidden Semi-Markov Model (CFHSMM) [7], additive FHMM [15] or Bayesian Nonparametric Hidden Semi-Markov Models [16]) to perform energy disaggregation. Still, there is an evident challenge to develop an accurate solution that could perform well for every type of appliance.

In this paper, the aim is to perform real-time flexibility detection using energy disaggregation techniques. Therefore, the key methodological contribution of this paper is a machine learning based tool for exploiting the building energy disaggregation capabilities in an online manner. Our contributions can

be summarized as follows. Firstly, we investigate the use of classification methods to perform energy disaggregation. Consequently, a comparison is performed between four widely-used classification methods, namely Naive Bayes (NB), K-Nearest Neighbors (KNN), Support Vector Machine (SVM) and AdaBoost. Secondly, we introduce a Restricted Boltzmann Machine (RBM) to perform automatic feature extraction in order to improve the performance of the four classification methods discussed. We validate our proposed approach by using a real measurement database, specifically conceived for energy disaggregation, i.e. the REDD [17].

The remaining of this paper is organized as follows. Section II introduces the problem description. Section III describes our proposed approach for the energy disaggregation problem. In Section IV the experimental validation of the proposed methods is detailed and Section V concludes the paper.

II. PROBLEM FORMULATION AND METHODOLOGY

This section details the problem definition targeted in this paper. In one unified framework, we split the problem into two parts, where first the energy disaggregation problem is solved, and then an identification procedure is carried out to analyze the potential of building demand flexibility.

The proposed solution for energy disaggregation is addressed using four different classification methods. More formally, let us define an input space $\mathcal{D}$ and an output space (label space) $\mathcal{B}$. The question of learning is reduced to the question of estimating a functional relationship of the form $\mathcal{C} : D \rightarrow B$, that is a relationship between inputs and outputs. A classification algorithm is a procedure that takes the training data as input and outputs a classifier $\mathcal{C}$. The goal is then to find a $\mathcal{C}$ which makes as few errors as possible. Intuitively, the learned classifier should be based on *enough* training examples, *fit* the training example and should be *simple*. Moreover, classification can be thought of as two separate problems: binary classification and multi-class classification.
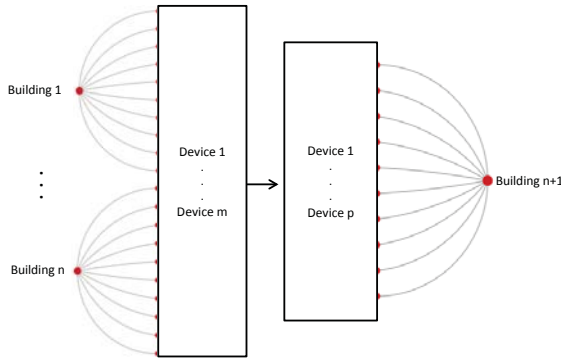

Fig. 1. Energy disaggregation

In our specific case, the B space is given by the electrical devices in the building, and the D space is given by the aggregated electrical energy consumption of the building. In Figure 1 the flow diagram of the energy disaggregation procedure is depicted. Firstly, using data from n buildings we derive a corresponding model for each device inside them. Furthermore these binary classification models are used to automatically classify, whether a given device is active at any specific moment in time, by using the building's total electrical energy consumption profile.

III. PROPOSED METHODS

In this section, we firstly briefly describe the four classification methods to perform energy disaggregation, these methods being part of the supervised learning paradigm. Secondly, we introduce the mathematical details of the Restricted Boltzmann Machine used to perform automatic features extraction, this method being part of the unsupervised learning paradigm.

A. Classification methods

For the classification problem, plenty of deterministic or probabilistic algorithms are known, where every observation is analyzed into a set of quantifiable properties, such as Naive Bayes [18], Support Vector Machine [19], AdaBoost [20], Random Forest Trees and so on. Prior studies tried to determine the most accurate classification method, as is shown in [21], but currently there is not a general consensus in the favor of a particular method.

1) Naive Bayes: is one of the most simple classification method based on a strong independence assumptions between the input features. Despite these relatively naive assumptions, with a training phase extremely easy to implement and fast computational time, Naive Bayes classifiers often outperform more sophisticated alternatives.

2) k-Nearest Neighbors: is a non-parametric method used for classification. The standard version of KNN used in this paper performs successively two steps. Specifically, the clusters are construct by partitioning the k-nearest neighbors based on a distance measure (i.e. Euclidean distance), followed by an update rule, such that the majority of those k-nearest neighbors decide the class of the next observations.

3) AdaBoost: it stands for Adaptive Boosting, and is a machine learning algorithm, which was proposed in the computational learning theory field by Y. Freund and R. Schapire [20]. AdaBoost method solves the classification problem using a linear combination of many weak classifiers into a single strong classifier. Acting as an expert, boosting often does not suffer from overfitting and it is worth to investigate in the context of our challenging dataset.

4) Support Vector Machine (SVM): is introduced by Vapnik in 1995 [19] and becomes very popular for solving problems in classification, regression, and novelty detection. An important characteristic of SVM is that the determination of the model parameters corresponds to a convex optimization problem, and so any local solution is also a global optimum. This guarantee comes with some computational cost but also with a better robustness.

B. Restricted Boltzmann Machine

Restricted Boltzmann Machine is a two-layer generative stochastic neural network which is capable to learn a probability distribution over its set of inputs [22]. Such a model

does not allow intra-layer connections between the units, and it allows just inter-layer connections. In fact, any unit from one layer has undirected connections to all the units from the other layers. Up to now, various types of restricted Boltzmann machines are already developed and successfully applied in different applications [23]. Despite their differences, almost all of these architectures preserve RBMs characteristics. To formalize a restricted Boltzmann machine, and its variants, three main ingredients are required, namely an energy function providing scalar values for a given configuration of the network, the probabilistic inference and the learning rules required for fitting the free parameters.

Thus, a RBM consists in two binary layers, the visible layer, $\mathbf{v} = [v_1, v_2, .., v_{n_v}]$, in which each neuron represents one dimension (feature) of the input data and the hidden layer, $\mathbf{h} = [h_1, h_2, .., h_{n_h}]$, which represents hidden features extracted automatically by the RBM model from the input data, where n_v is the number of visible neurons and n_h is the number of the hidden neurons. Each visible neuron i is connected to any hidden neuron j by a weight, i.e. W_{ij}. All these weights are stored in a matrix $\mathbf{W} \in \mathcal{R}^{n_v \times n_h}$, where $\mathcal{R}$ is the set of real numbers, in which the rows represent the visible neurons and the columns the hidden ones. Finally, each visible neuron i has associated a bias a_i which is stored in a vector $\mathbf{a} = [a_1, a_2, .., a_{n_v}]$. Similarly, the hidden neurons have biases which are stored in a vector $\mathbf{b} = [b_1, b_2, .., b_{n_h}]$. Further on, we will note with $\mathbf{\Theta} = \{\mathbf{W}, \mathbf{a}, \mathbf{b}\}$ a set which represent the union of all free parameters of a RBM (i.e. weights and biases). Formally, the energy function of a RBM for any state $\{\mathbf{v}, \mathbf{h}\}$ can be computed by summing over all possible interactions between neurons, weights and biases, as folows:

$$E(v, h) = -\sum_{i=1}^{n_v}\sum_{j=1}^{n_h} v_i h_j W_{ij} - \sum_{i=1}^{n_v} v_i a_i - \sum_{j=1}^{n_h} h_j b_j \quad (1)$$

where the term $\sum_{i=1}^{n_v}\sum_{j=1}^{n_h} v_i h_j W_{ij}$ is given by the total energy between the neurons from different layers, while $\sum_{i=1}^{n_v} v_i a_i$ represents the energy of the visible neurons and $\sum_{j=1}^{n_h} h_j b_j$ is the energy of the hidden neurons.

The inference in a RBM means to determine two conditional distributions. For any hidden or visible neuron this can be done just by sampling from a sigmoid function, as shown below:

$$p(h_j = 1|\mathbf{v}, \mathbf{\Theta}) = \frac{1}{1 + e^{-(b_j + \sum_{i=1}^{n_v} v_i w_{ij})}} \quad (2)$$

$$p(v_i = 1|\mathbf{h}, \mathbf{\Theta}) = \frac{1}{1 + e^{-(a_i + \sum_{j=1}^{n_h} h_j w_{ij})}} \quad (3)$$

To learn the parameters of a RBM model there are more variants in the literature (e.g. persistent contrastive divergence, parallel tempering [24], fast persistent contrastive divergence [25]). Almost all of them being derived from the Contrastive Divergence (CD) method proposed by Hinton in [26]. For this reason, in this paper, we briefly describe and use just the original CD method. CD is an approximation of the maximum likelihood learning, which is practically intractable in a RBM. Thus, while in maximum likelihood the learning

phase minimizes the Kullback-Leiber (KL) measure between the distribution of the input data and the model approximation, in CD the learning follows the gradient of:

$$CD_n \propto D_{KL}(p_0(\mathbf{x})||p_\infty(\mathbf{x})) - D_{KL}(p_n(\mathbf{x})||p_\infty(\mathbf{x})) \quad (4)$$

where, $p_n(.)$ represents the resulting distribution of a Markov chain running for n steps. Furthermore, the general update rule of the free parameters of a RBM model is given by:

$$\Delta\mathbf{\Theta}_{\tau+1} = \rho\Delta\mathbf{\Theta}_\tau + \alpha(\nabla\mathbf{\Theta}_{\tau+1} - \xi\mathbf{\Theta}_\tau) \quad (5)$$

where τ, α, ρ, and ξ represent the update number, learning rate, momentum, and weights decay, respectively, as thoroughly discussed in [27]. Moreover, $\nabla\mathbf{\Theta}_{\tau+1}$ for each free parameter may be computed by deriving the energy function from Equation 1 with respect to that parameter, as detailed in [26], yielding:

$$\nabla w_{ij} \propto \langle v_i h_j \rangle_0 - \langle v_i h_j \rangle_n \quad (6)$$

$$\nabla a_i \propto \langle v_i \rangle_0 - \langle v_i \rangle_n \quad (7)$$

$$\nabla b_j \propto \langle h_j \rangle_0 - \langle h_j \rangle_n \quad (8)$$

with $\langle \cdot \rangle_n$ being the distribution of the model obtained after n steps of Gibbs sampling in a Markov Chain which starts from the original data distribution $\langle \cdot \rangle_0$.

IV. EXPERIMENTAL RESULTS

In this section we analyze and validate our proposed approach using a real-world database, namely *The Reference Energy Disaggregation Dataset* (REDD), described by Kolter and Johnson in [17]. This data was chosen as it is an open dataset[1] collected specifically for evaluating energy disaggregation methods. It contains aggregated data recorded from six buildings over few weeks sampled at 1 second resolution, together with the specific data for all appliances of each building at 3 seconds resolution.

In the first set of experiments, we study the performance of the classification methods (i.e. Naive Bayes, K-Nearest Neighbors, Support Vector Machine and AdaBoost) for detecting the activation of four appliances (i.e. refrigerator, electric heater, washer-dryer, dishwasher), specifically chosen for their ability to provide demand-side flexibility. Furthermore, in the second stage we demonstrate the improvement in the accuracy of the classification after a Restricted Boltzmann Machine is used for automatic feature extraction. Finally, assuming the aforementioned four appliances shiftable in time, we discuss the possible benefits of real-time flexibility detection.

The experiments were performed in the MATLAB® environment using the methods described in Section III. For the classification methods we have used the optimized parameters from the machine learning toolbox (e.g. SVM with radial kernel function). For each appliance we have built a separate binary classification model for every classification method. The input at every moment in time is given by a window of 10 consecutive time steps from the aggregated building consumption, while the output was represented by the activation of the

---

[1]http://redd.csail.mit.edu/, Last visit November 5th, 2015

appliance (i.e. on/off status). In all the experiments performed, we have trained the models on 5 buildings (i.e. 2, 3, 4, 5, and 6) and we have tested the models on a different building (i.e. 1). Also, as recommended in [14], we have applied a median filter of 6 samples to make the data smoother.

For the feature extraction procedure we have implemented RBMs with the following parameters: 20 hidden neurons and 10 visible neurons (representing the time window of 10 consecutive time steps). After a short fine tuning procedure, the learning rate was set to 10^{-2}, the momentum was set to 0.5, and the weight decay was set to 0.0002. We trained the RBM models for 25 epochs, and after that we have used the probabilities of the hidden neurons as inputs for the classification methods.

In order to characterize as fairly as possible the accuracy of the models proposed to classify the appliance activation we have calculated the classifier accuracy as follows:

$$Accuracy = \frac{\sum_{i=1}^{n} A_{ii}}{\sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij}} \quad (9)$$

where A is the confusion matrix (also known as a contingency table or an error matrix), A_{ii} represents the positive true value and the denominator represents the total number of data used in the classification procedure. This quantifies the proportion of the total number of instances that were correctly classified.

A. Energy disaggregation

In this subsection, we first perform a comparison between the four classification methods, namely Naive Bayes (NB), k-Nearest Neighbors (KNN), Support Vector Machine (SVM) and AdaBoost (AB). Table I summarizes the classification accuracy for different building electrical components, such as refrigerator, electric heater, washer-dryer and dishwasher. For a better insight into the results, an example of the energy consumption for the appliances corresponding to building 1 (the test data) is depicted in Figure 3.

TABLE I
RESULTS SHOWING ACCURACY [%] FOR EACH OF NAIVE BAYES, KNN, SVM AND ADABOOST TO CLASSIFY AN APPLIANCE VERSUS ALL DATA.

Appliance	NB	KNN	SVM	AdaBoost
refrigerator	52.18%	67.36%	67.45%	**87.13%**
electric heater	93.01%	97.79%	**98.84%**	94.74%
washer dryer	92.04%	**96.17%**	78.27%	95.56%
dishwasher	97.52%	**98.11%**	97.74%	97.77%

Furthermore, to improve the classification performance, we have employed the automatic features extraction procedure by using the Restricted Boltzmann Machine as described in SectionIII-B. Next, the extracted features are used as inputs for the classification methods. We have tested and validated this approach on the same electrical appliances as before, as shown in Table II. It can be observed that in all situations, the use of RBMs has improved the accuracy for each classifier. This culminates with an improvement of around 30% for the case of the refrigerator classified with KNN, from 67.36% initial accuracy, up to 96.72% accuracy after the use of RBM.
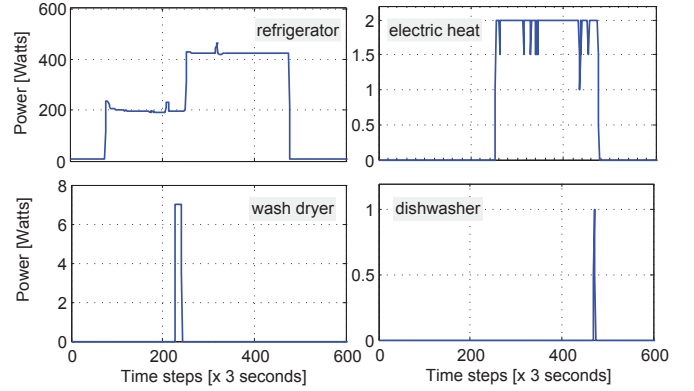

Fig. 2. An example of energy consumption in Building 1 over 30 minutes for refrigerator, electric heater, washer dryer and dishwasher.

TABLE II
RESULTS SHOWING ACCURACY [%] FOR EACH OF NAïVE BASE, KNN, SVM AND ADABOOST WITH RBM EXTENSION, TO CLASSIFY AN APPLIANCE VERSUS ALL DATA.

Appliance	NB-RBM	KNN-RBM	SVM-RBM	AB-RBM
refrigerator	64.78%	**96.72%**	84.45%	91.02%
electric heater	99.13%	99.81%	**99.86%**	99.84%
washer dryer	99.14%	97.31%	89.23%	**99.27%**
dishwasher	97.64%	98.43%	**98.67%**	97.82%

It is worth mentioning, that the imbalanced number of data points in every class suggests that a more deeper data mining analysis may be useful. In term of computational complexity the training time varies from the range of few seconds in the case of KNN up to few minutes in the case of SVM. In the testing phase, to classify all the data points considered (i.e. 745868 instances per year per appliance) each of the methods has ran in approximately 1 second, except SVM which ran in 4-5 seconds. Overall, this yields an execution time of a few microseconds per data point making the approach suitable for a large range of real-time applications.

B. Flexibility detection

The energy disaggregation results may be used further in a large number of applications, as reported in 2015 by the US Department of Energy in an extensive report [28] which aims to characterize the actual performance of energy disaggregation solutions used in both the academic research and in commercial products.

Most importantly, our results may be used to detect in real-time the building flexibility available. We observed that approximately 17% of the total energy consumption for building 1 is used by the four disaggregated appliances, such as refrigerator 11.72%, electric heater 5.08%, washer-dryer 0.0007% and dishwasher 0.9% respectively. More statistical details about these appliances for building 1 are presented in Table 3. A visual examination of the results, assuming that all the four appliances studied have smart time-shifting capabilities, and a detection accuracy of over 96% in all the experiments, show a significant peak reduction. As by example, in Figure 3 the inflexible load is represented by the

TABLE III
GENERAL CHARACTERISTICS OF THE BUILDING 1 APPLIANCES USED IN
THE EXPERIMENTS.

	Mean	Standard deviation
refrigerator	56.41	86.65
electric heater	24.44	148.16
wash dryer	0.11	0.96
dishwasher	4.30	43.54

difference between the total energy consumption signal and the sum of our disaggregated signals over 24 hours. In this case, we may observe that the average buildings flexibility is 23.21%.
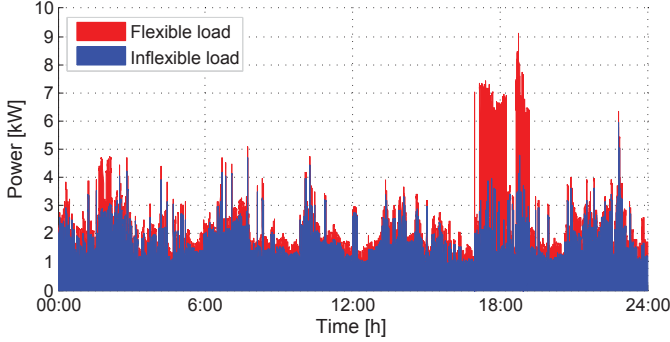

Fig. 3. An example of electrical energy consumption in buildings over one day for inflexible load and flexible load (refrigerator, electric heater, washer dryer and dishwasher).

V. CONCLUSION

In this paper a novel tool capable to perform accurate energy disaggregation for real-time flexibility detection is proposed. A comparison between four existing classification methods was performed. Aiming at enhancing the quality of such estimates as well as at increasing the accuracy of energy disaggregation, a method for automatic features extraction is proposed, using Restricted Boltzmann Machines. By incorporating the RBM for feature extraction, each of the classification methods, i.e. Naive Bayes, k-Nearest Neighbors, Support Vector Machine and AdaBoost, has outperformed its non-preprocessed counterpart. The experimental validation performed on the REDD dataset shows that KNN- RBM has the best trade-off between accuracy and speed.

ACKNOWLEDGMENT

REFERENCES

[1] P. Nejat, F. Jomehzadeh, M. M. Taheri, M. Gohari, and M. Z. A. Majid, "A global review of energy consumption, co_2 emissions and policy in the residential sector," *Renewable and Sustainable Energy Reviews*, vol. 43, pp. 843 – 862, 2015.

[2] E. Kara, Z. Kolter, M. Berges, B. Krogh, G. Hug, and T. Yuksel, "A moving horizon state estimator in the control of thermostatically controlled loads for demand response," in *IEEE International Conference on Smart Grid Communications*, Oct 2013, pp. 253–258.

[3] G. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, Dec 1992.

[4] F. Sultanem, "Using appliance signatures for monitoring residential loads at meter panel level," *IEEE Transactions on Power Delivery*, vol. 6, no. 4, pp. 1380–1385, Oct 1991.

[5] C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, L. Norford, and P. Armstrong, "Power signature analysis," *IEEE Power and Energy Magazine*, vol. 1, no. 2, pp. 56–63, Mar 2003.

[6] M. A. Guvensan, Z. C. Taysi, and T. Melodia, "Energy monitoring in residential spaces with audio sensor nodes: Tinyears," *Ad Hoc Networks*, vol. 11, no. 5, pp. 1539 – 1555, 2013.

[7] H. Kim, M. Marwah, M. Arlitt, G. Lyon, and J. Han, "Unsupervised disaggregation of low frequency power measurements," in *SIAM International Conference on Data Mining*, pp. 747–758.

[8] J. Z. Kolter, S. Batra, and A. Y. Ng, "Energy disaggregation via discriminative sparse coding," pp. 1153–1161, 2010.

[9] Y. Du, L. Du, B. Lu, R. Harley, and T. Habetler, "A review of identification and monitoring methods for electric loads in commercial and residential buildings," in *IEEE Energy Conversion Congress and Exposition*, Sept 2010, pp. 4527–4533.

[10] M. Zeifman and K. Roth, "Nonintrusive appliance load monitoring: Review and outlook," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 1, pp. 76–84, February 2011.

[11] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," *Sensors*, vol. 12, no. 12, p. 16838, 2012.

[12] J. Kelly and W. Knottenbelt, "Metadata for energy disaggregation," in *IEEE 38th International Computer Software and Applications Conference Workshops*, July 2014, pp. 578–583.

[13] D. Bergman, D. Jin, J. Juen, N. Tanaka, C. Gunter, and A. Wright, "Distributed non-intrusive load monitoring," in *IEEE PES Innovative Smart Grid Technologies*, Jan 2011, pp. 1–8.

[14] A. Iwayemi and C. Zhou, "Leveraging smart meters for residential energy disaggregation," in *IEEE PES General Meeting — Conference Exposition*, July 2014, pp. 1–5.

[15] J. Z. Kolter and T. Jaakkola, "Approximate inference in additive factorial hmms with application to energy disaggregation," *Journal of Machine Learning Research - Workshop and Conference Proceedings*, vol. 22, pp. 1472–1482.

[16] M. J. Johnson and A. S. Willsky, "Bayesian nonparametric hidden semi-markov models," *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 673–701, Feb. 2013.

[17] J. Z. Kolter and M. J. Johnson, "REDD: A Public Data Set for Energy Disaggregation Research," 2011.

[18] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, 1st ed.   Springer, Oct. 2007.

[19] C. Cortes and V. Vapnik, "Support-Vector Networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[20] Y. Freund and R. E. Schapire, "A short introduction to boosting," in *In Proceedings of the 60th International Joint Conference on Artificial Intelligence*.   Morgan Kaufmann, 1999, pp. 1401–1406.

[21] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML '06, 2006, pp. 161–168.

[22] P. Smolensky, "Information processing in dynamical systems: Foundations of harmony theory," in *Parallel Distributed Processing: Volume 1: Foundations*, 1987.

[23] E. Mocanu, P. H.Nguyen, M. Gibescu, and W. Kling, "Comparison of machine learning methods for estimating energy consumption in buildings," in *Proceedings of the 13th International Conference on Probabilistic Methods Applied to Power Systems, Durham, UK*, 2014.

[24] G. Desjardins, A. Courville, Y. Bengio, P. Vincent, and O. Delalleau, "Tempered Markov Chain Monte Carlo for training of restricted Boltzmann machines," in *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, pp. 145–152.

[25] T. Tieleman and G. Hinton, "Using fast weights to improve persistent contrastive divergence," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML, 2009, pp. 1033–1040.

[26] G. E. Hinton, "Training Products of Experts by Minimizing Contrastive Divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, Aug. 2002.

[27] G. Hinton, "A Practical Guide to Training Restricted Boltzmann Machines," Tech. Rep., 2010.

[28] E. Mayhorn, G. Sullivan, R. Butner, H. Hao, and M. Baechler, "Characteristics and performance of existing load disaggregation technologies," in *PNNL-24230*, 2015.