A Learning Framework for Control-Oriented Modeling of Buildings

Conference Paper · December 2017
DOI: 10.1109/ICMLA.2017.00079

A Learning Framework for Control-Oriented Modeling of Buildings

Javier Rubio-Herrero
Department of Engineering,
St. Mary's University
San Antonio, Texas 78228
Email: jrubioherrero@stmarytx.edu

Vikas Chandan, Charles M Siegel, Abhinav Vishnu, Draguna L Vrabie
Pacific Northwest National Laboratory (PNNL)
Richland, Washington 99352
Email: {vikas.chandan, charles.siegel, abhinav.vishnu, draguna.vrabie}@pnnl.gov

***Abstract*—Buildings consume almost 40% of energy in the US. In order to optimize the operation of buildings, models that describe the relationship between energy consumption and control knobs such as set-points with high predictive capability are required. Data driven modeling techniques have been investigated to a somewhat limited extent for optimizing the operation and control of buildings. In this context, deep learning techniques such as Recurrent Neural Networks (RNNs) hold promise, empowered by advanced computational capabilities and big data opportunities. This paper investigates the use of deep learning for modeling the power consumption of building heating, ventilation and air-conditioning (HVAC) systems. A preliminary analysis of the performance of the methodology for different architectures is conducted. Results show that the proposed methodology outperforms other data driven modeling techniques significantly.**

I. Introduction

Buildings account for almost 40% of the energy consumption in the U.S [1]. In the last couple of decades, various stakeholders such as building energy managers, utility companies, policy-makers, and researchers have identified, demonstrated, and advocated several solutions to achieve the goals of energy and cost efficiency in buildings. Some of these solutions include demand response (DR) [2], pre-cooling or pre-heating [3], optimal supervisory control of underlying systems such as heating ventilation and air-conditioning (HVAC) [4], and on-site renewables [5]. However, these solutions require a mathematical model of energy consumption as an essential element.

Building energy consumption modeling is a complex problem because of its dependency on a large number of factors such as set-point temperatures, weather, occupant behavior, underlying control systems, building layout, and equipment efficiencies. A classic approach to solving this problem is to retrieve insights from physics-based models to describe system dynamics. For example, EnergyPlus [6] is a widely physics based modeling tool that is widely used. However, the main limitation of these models is that they involve solving equations with physical parameters that are specific to the buildings at hand and therefore there is usually a great effort spent on gathering sufficient information about the physical features of the environment (building materials, heat transfer constants, boundary conditions, etc.), which can be time consuming and expensive to obtain. In this context, the availability of data at various spatial and temporal granularities due to the rapid penetration of information technology (IT) in today's buildings [7] provides an opportunity to address the challenges associated with physics based models. Therefore, data driven modeling paradigms such as linear regression, support vector regression, and artificial neural networks are finding application in the buildings domain [1].

In order to drive solutions related to energy and cost efficiency of building operations, it is imperative that the underlying models be *control oriented*. In other words, they should have the ability to quantify the energy or cost of energy as a function of control knobs such as set-point temperatures, ON/OFF status of devices, etc. in addition to exogenous factors such as weather and occupancy. Also, the models should have satisfactory predictive capability, impose minimal data pre-processing requirements, and have the ability to be adapted continuously to account for changing conditions as new data becomes available. In this regard, deep learning techniques such as Recurrent Neural Networks (RNNs) [8], empowered by advanced computational capabilities and big data opportunities, hold promise. While RNNs have been somewhat investigated in a small number of prior publications (Section II), their full potential to provide control oriented and accurate models of building energy is still untapped. We seek to address this gap by demonstrating the potential of RNNs to provide models that satisfy all the requirements mentioned earlier and enable multiple control use cases. The modeling approach is tested on data from a real building. Our results show that the proposed methodology outperforms other data driven modeling techniques with modeling errors that are 8.5-52% lower. We also perform a preliminary analysis of the RNNs with regards to depth and breadth of the network, and sensitivity to the outside temperature.

The paper is organized as follows: a related work summary is provided in Section II; Section III describes the RNN based framework used in the paper for building energy modeling; Section IV describes the data handling processes involved; results obtained from a real world building data-set are discussed in Section V; Section VI summarizes the conclusions

Javier Rubio-Herrero was at PNNL when this research was conducted.

and discusses avenues for future research.

II. Related Work

Data driven techniques have been applied in the context of building energy modeling. Several tools have been investigated in this context such as simple and multiple linear regression, artificial neural networks (ANNs), and support vector regression (SVR) [9]–[12] for various objectives such as to help understand the energy consumption as a function of a reduced number of variables, usually related to weather or building materials. Researchers have also investigated how to characterize the thermal characteristics of a building and include that information in cost-minimization control problems [13], [14]. Kalman filters or Markov models have been used to this end [15]. Neural networks have also been studied to estimate the thermal properties of a building [16].

As previously mentioned, the availability of large data streams from today's building automation and management systems provides an opportunity to strengthen our capability to model and optimize a building's behavior. This can be done by applying advanced *deep learning* techniques that can transform these big data streams and other sources of data into recommendations for energy and cost minimization with optimal control. There have been a few recent developments in the application of deep learning for energy prediction. Google recently implemented neural networks to model power use efficiency (PUE) in their data centers [17]. Shallow recurrent neural networks with very few features have also been tested to predict heating gas consumption [18]. Conditional Restricted Boltzmann Machines and Factored Conditional Restricted Boltzmann Machines have been applied to forecast energy loads in buildings [19], [20]. Deep recurrent neural networks have been recently designed to tackle energy disaggregation problems, wherein individual consumption of appliances are estimated from a single measurement [21]. Researchers have also developed unsupervised methods for identifying the relationships between sensor data streams in buildings [22], [23].

Generally speaking, the data collected from buildings can be classified as

- **Controls:** human-provided operating inputs that at any given time are set either using a pre-programmed logic or manually by the building manager. Examples are zone set-point temperatures, duct static pressure set-point, and mode of operation of HVAC (heating/cooling).
- **Exogenous inputs:** inputs such as outside weather and zone occupancy status, which drive the energy consumption in a building, but cannot be controlled.
- **Internal variables:** operating conditions in the building that are a consequence of the choice of controls, such as zone temperatures, air flow rates in HVAC systems, return and supply temperatures of water and air, etc.
- **Performance variables:** measurements indicative of performance such as heating and cooling energy consumption, active and reactive power consumption, etc.

The focus of this article is to investigate the use of RNNs for modeling building HVAC energy consumption to enable control applications such as set-point optimization. Based on related work surveyed in this section, the potential of RNNs towards meeting such goals has not been well investigated. A recent article [24] uses RNNs for building energy consumption modeling, where the underlying architecture involves predicting energy consumption at future time instances based on energy consumption at the current time instance. On the other hand, our work augments and demonstrates the capability of RNNs for the broader objective of modeling performance variables and internal variables as a function of controls and exogenous inputs.

III. Modeling methodology

The modeling problem that we consider in this work involves predicting the total building HVAC thermal demand as well as zone temperatures as a function of control variables and exogenous inputs. Our proposed modeling process consists of two steps: (i) data handling, (represented schematically in Figure 3), and (ii) deep learning. The first step is explained in the next section. In this section, we focus on the deep learning aspect of the modeling process.

Once the data is cleaned and organized as a time series (c.f. Section IV), we proceed with applying RNN to model the outputs as a function of the inputs described above. RNNs are specially suitable to address prediction and classification problems in which the inputs are expressed as time series. In particular, they appear promising in the context of modeling a dynamical system such as a building because of their superior ability to capture nonlinear and dynamic dependencies, when compared to other machine learning techniques. Their structure differs from that of multilayer perceptrons in that the information that belongs to a time stamp is fed back to the neural network, and thus this information is taken into account when updating the weights of the neural network. This makes the model learn about the temporal dependence between the inputs and the outputs (see Figure 1). However, RNNs may suffer from a *vanishing gradient problem*; this problem is a consequence of small gradients being back propagated during the training phase in deep neural networks and results in negligible updates of the weights in earlier layers of the network. An RNN architecture called *long short-term memory* (LSTM) provides the state-of-the-art solution to this problem. An LSTM unit presents an input gate, a forget gate, and an output gate (see Figure 2) that are able to learn both long and short time relations while circumventing the appearance of vanishing gradients [25]. In this work we use RNNs with LSTM units.

IV. Data handling

The data handling process consists of the steps shown in Figure 3. We explain the process in the context of the data set available for the building described in the next section. However, these underlying steps can also be easily implemented on other data sets.

The data streams needed for the model described in the above section are downloaded from a central database. For the

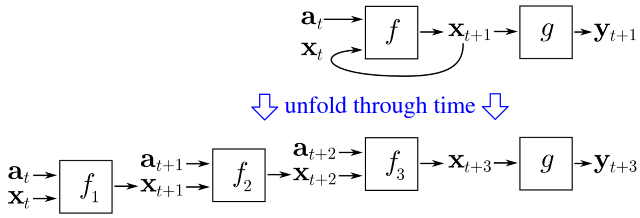

Fig. 1. General scheme of a RNN.

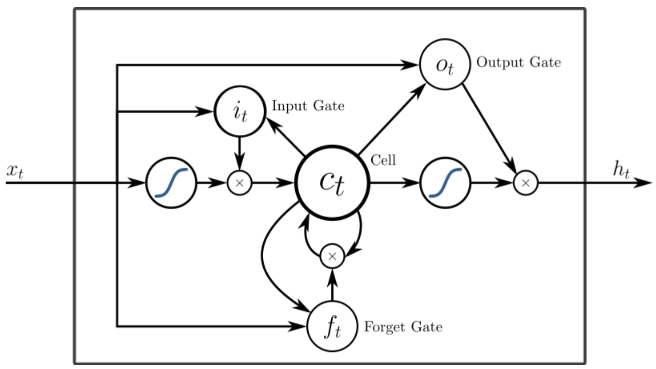

Fig. 2. General scheme of a LSTM. Adapted from [26].

specific building described in Section V, data for each variable or topic was collected every minute and downloaded as a *.csv* file with three columns, namely, time stamp, value, and units. The result of this step was therefore a list of as many .csv files as topics.

The next steps involve data categorization and cleaning. In categorizing the data streams, we reorganize our list of .csv files into four larger .csv files according to the classification previously mentioned (performance variables, internal variables, controls, and external inputs). Each of those new files has as many columns as variables correspond to that specific category, plus one, corresponding to time stamps.

The subsequent step is data cleaning, where data gaps are addressed. Depending on the length of data gap, we consider three possible scenarios.

1) **Data gaps that are scattered through the data set and do not last for more than a few minutes:** Data rows where such gaps appear can be directly dropped. In other words, time stamps where such gaps appear are deleted across the data set.
2) **Gaps that extend for no more than a few consecutive hours:** These can be addressed by filling the gaps with reasonable values like the mean or the median of the topics in question.
3) **Gaps that extend from a few hours up to several days:** These are removed from the data set. Recent techniques such as *phased LSTMs* [27], adapt the rate of learning in the event of such gaps, claiming an accelerated rate of convergence. However, such techniques were not considered because convergence was observed to be 'fast' (less than a minute) even with regular LSTM units for the modeling problem in Section V.

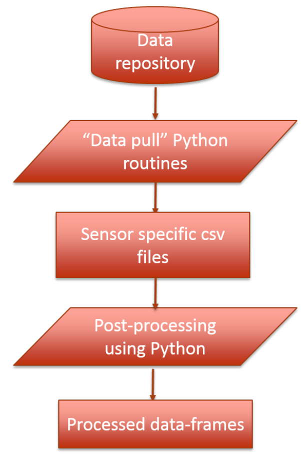

Fig. 3. Data processing steps

The last step in data postprocessing involves removal of outliers by a *data smoothening* process where we replace spiky measurements with a moving average of measurements over the previous 10 periods. This process is represented in (1), where a spiky measurement of X occurs at time-stamp T_s and is replaced by $\hat{X}(T_s)$.

$$\hat{X}(T_s) = \frac{\sum_{k=T_s-9}^{k=T_s} X(k)}{10} \tag{1}$$

V. Implementation Details

In this section, we provide implementation details of the RNN based modeling framework described in Section III on a real world building, for which data along both spatial and temporal dimensions is available. We assess various aspects of the modeling process as applied to this building.

A. Building and data description

The building used in this study is a newly constructed office building. Heating and cooling are delivered by a variable air volume (VAV) system served by 4 air handling units (AHUs) serving 24 VAV boxes (zones). Each VAV box is equipped with a hot water reheat coil. A boiler, fed by natural gas, supplies hot water to the reheat coils and AHU coils. Chilled water is supplied by a central chiller plant.

Data from specific sensors for the above-mentioned buildings is stored in a database, which communicates with the building management system (BMS) and polls data for these sensors at a time resolution of 1 minute. A total of 667 sensors report data corresponding to measurements such as supply and return temperatures of air and water, air, hot water and cold water flow rates, energy and power consumption, set-points for the underlying control systems, occupancy status in zones, and outside air temperature.

Data handling was performed using the steps mentioned in Section IV. We used around 8 months of data corresponding to July 30, 2016 to March 20, 2017. The modeling problem corresponds to predicting the thermal demand (heating and cooling) at the building level and the zone internal temperatures based on the variables in the exogenous inputs and controls category as inputs. The number of input channels was 198; the number of output channels was 26, which consisted of total building heating demand, total building cooling demand and 24 zone temperatures. The heating demand (Q_H) and cooling demand (Q_C) of the building were obtained using equations (2) and (3). Here, $\dot{m}_{HW}$ and $\dot{m}_{CHW}$ represent the mass flow rates of hot and cold water streams, respectively, supplied to the building, and ΔT_{HW} and ΔT_{CHW} represent the temperature difference (between return and supply streams) of hot and cold water respectively; c_p is the specific heat capacity of water, assumed constant at 4.18 kJ/kg-K.

$$Q_H = c_p \dot{m}_{HW} \Delta T_{HW} \tag{2}$$

$$Q_C = c_p \dot{m}_{CHW} \Delta T_{CHW} \tag{3}$$

All values of input and output channels were normalized by subtracting from them the mean value and then dividing by one standard deviation. The data used for validation was chosen to be the union of the last 20% of the first 30% of the entire data-set, and the last 20% of the remaining 70% of the data-set. This choice of the validation data covers the last few weeks of both summer and winter.

B. Modeling

In this section we perform a preliminary assessment of the RNN based modeling methodology applied to the above-mentioned building. We first undertake a topology assessment, where we investigate the impact of the breadth and depth of the network - via an exhaustive search on a limited search space - on the modeling errors and identify a topology (referred to as optimal topology in the rest of the paper) that results in the smallest error among all topologies considered. We then perform a bench-marking analysis where we compare the performance of the RNN with optimal topology with other black box modeling techniques.

1) Topology analysis: The RNN was built with *Keras*, a Python package [28]. The optimizer *RMSprop* was used to minimize the mean square error (MSE) between prediction and ground truth for the outputs. 19 topologies, as shown in Table I were compared. Each topology is defined by two parameters - number of hidden layers which represents the network depth, and the number of nodes per layer which represents the layer breadth. The number of hidden layers in these topologies vary from 1 to 3, and the number of LSTM units (nodes) in each layer is either 32, 64 or 128. We used *PReLU* activation [29] for all hidden layers and linear activation for output layer. We observe that a single hidden layer with 128 nodes provides the smallest validation error. Since the number of layers is a measure of the non-linearity present in the system dynamics. this suggests that the system is somewhat weakly nonlinear.

2) Bench-marking: To compare the RNN with other approaches, we also applied other machine learning techniques on the post-processed data using the same training and testing time windows as above. The approaches implemented are Multilayer Perceptron (MLP), Linear Regression (LinReg), Linear Support Vector Regression (LinSVR), and Random Forest (RF) [30]. Each of these approaches were implemented in Python using the *Scikit-Learn* package, except MLP which was implemented using *Keras*. The parameters of all these methodologies were tuned, through an exhaustive search, so that the modeling error was small. MLP was implemented with two hidden layers. Similar to the RNN approach, *PReLU* activation was used for all layers except the output layer, which used linear activation. LinReg used l_2 *regularization* parameter equal to 200. LinSVR and NonlinSVR used parameters $C = 1$ and $\epsilon = 0.025$. RF was implemented with 15 trees in the forest. We used MSE over the test data-set as the appropriate metric to compare the performance of RNN with the above-mentioned machine learning approaches, as shown in Figure 4. The RNN architecture used for comparison with the other approaches had a single layer with 128 nodes that minimized the validation MSE in Table I. Note that the MSE is dimensionless because it is computed using normalized data. It predicts the outputs as a function of inputs for one time step at a time. We observed that RNN clearly outperformed all the other techniques, with 12.8 - 24.3% and 8.5 - 51.8% lower MSE respectively for heating and cooling demand predictions, evaluated over the validation data-set. A comparison of the predicted and measured zone temperatures (for one of the zones in the building) and overall building cooling demand, for a subset of the validation dataset is presented in Fig. 5. From these plots, we observe that the predictions from the RNN based modeling framework align well with ground truth measurements.

TABLE I
RNN TOPOLOGY ANALYSIS

Topology	Training MSE	Testing MSE
(128)	0.006	0.031
(64)	0.007	0.034
(128, 128)	0.006	0.041
(64, 64)	0.007	0.044
(128, 128, 128)	0.006	0.049
(64, 128)	0.008	0.052
(32, 128)	0.012	0.06
(64, 64, 128)	0.009	0.06
(64, 64, 64)	0.008	0.06
(64, 128, 128)	0.008	0.062
(32)	0.013	0.063
(32, 32)	0.015	0.067
(32, 128, 128)	0.01	0.072
(32, 64)	0.013	0.073
(32, 32, 64)	0.015	0.079
(32, 32, 128)	0.014	0.083
(32, 64, 128)	0.015	0.087
(32, 32, 32)	0.018	0.088
(32, 64, 64)	0.017	0.093

Note that the MSE in 4 is for heating and cooling demand predictions only, while the MSE reported in Table I includes all outputs - heating and cooling demands and zone temperatures. Since the latter MSE values are much smaller than the former, this suggests that the prediction of zone temperatures has smaller error than the prediction of thermal demands. This can also be observed in Figure 5.

In order to quickly verify that the RNN based modeling framework is consistent with physical laws governing the

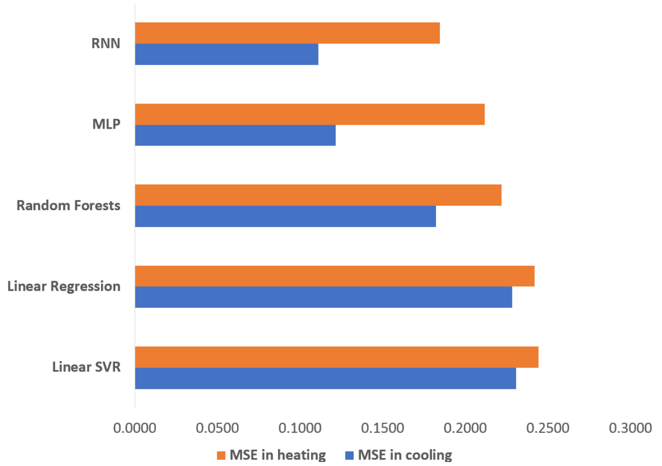

Fig. 4. Performance comparison of various machine learning methods against RNN over the validation data-set. The length of the bars denotes MSE

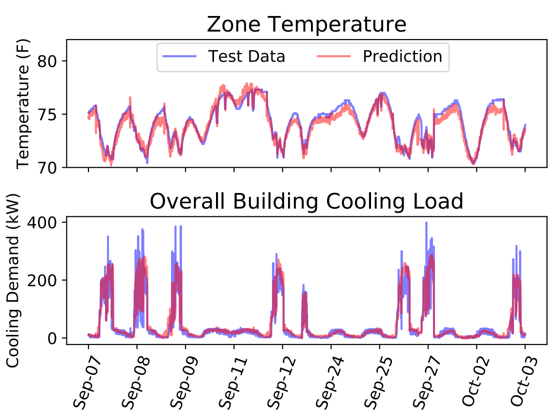

Fig. 5. Predicted vs. measured zone temperature and cooling demand using RNNs

TABLE II
RNN PRELIMINARY SENSITIVITY ANALYSIS

MSE	Zone 133	Zone 136
With external temperature	0.0213	0.015
Without external temperature	0.0231	0.0209
Change (%)	8.16	38.82

thermal dynamics of the building, we undertake a preliminary sensitivity analysis, wherein we quantify the impact of external temperature as an input on the accuracy of zone temperature predictions. In particular, we choose two zones - an externally facing zone (zone 136) and an internal zone (zone 133). We record the change in MSE on the validation data-set for temperature predictions of these zones, with and without the external temperature as an input. The results are shown in Table II, where we observe that the difference in MSE for the external zone is 38.82%, while for the internal zone it is only 8.16%. This verifies that the external zone is much sensitive to the outside temperature than the internal zone, as expected.

C. Discussion

We now present a summary the findings based on the above investigation. Firstly, we observed that the breadth of the network is more important than the depth in capturing the dynamics accurately. However, this was based on exploration using a single data-set. Further investigation using more data-sets is required in the future to conclusively establish the most appropriate architecture for modeling the thermal dynamics of buildings. Secondly, RNNs prove to be an effective tool for modeling the thermal demand, with significantly lower errors compared to other black-box methods. Lastly, a quick analysis reveals that the model appears consistent with the underlying physics.

VI. CONCLUSIONS

In this paper, we provided a preliminary investigation on the use of deep learning for modeling thermal demand and zone temperatures using a data-set collected from a building site. Results showed that the proposed methodology using recurrent neural networks (RNNs) outperforms other data driven modeling techniques. While some preliminary findings were obtained with regard to the architecture of the network to minimize the prediction errors, a more thorough investigation on additional data-sets is required as an area of future research to conclusively establish architecture types that are well suited for this modeling problem. Also, the modeling was performed on data-set from one building. It will be interesting to explore if the knowledge obtained by training models on one data-set can be used for developing models for other buildings of similar type. The applicability of the modeling framework to minimize energy and cost requirements is also a future direction for research. Appropriate use cases need to be developed to achieve such objectives.

VII. ACKNOWLEDGMENT

This work was supported by U.S. Department of Energy (contract DE-AC02-76RL01830) via. PNNL Laboratory Directed Research and Development (LDRD) funding under the Control of Complex Systems Initiative.

REFERENCES

[1] H.-x. Zhao and F. Magoulès, "A review on the prediction of building energy consumption," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 6, pp. 3586–3592, 2012.

[2] P. Palensky and D. Dietrich, "Demand side management: Demand response, intelligent energy systems, and smart loads," *IEEE transactions on industrial informatics*, vol. 7, no. 3, pp. 381–388, 2011.

[3] P. Xu, P. Haves, M. A. Piette, and J. Braun, "Peak demand reduction from pre-cooling with zone temperature reset in an office building," *Lawrence Berkeley National Laboratory*, 2004.

[4] S. Wang and Z. Ma, "Supervisory and optimal control of building HVAC systems: A review," *HVAC&R Research*, vol. 14, no. 1, pp. 3–32, 2008.

[5] S. Ruangpattana, D. Klabjan, J. Arinez, and S. Biller, "Optimization of on-site renewable energy generation for industrial sites," in *Power Systems Conference and Exposition (PSCE), 2011 IEEE/PES*. IEEE, 2011, pp. 1–8.

[6] D. B. Crawley, L. K. Lawrie, F. C. Winkelmann, W. F. Buhl, Y. J. Huang, C. O. Pedersen, R. K. Strand, R. J. Liesen, D. E. Fisher, M. J. Witte *et al.*, "Energyplus: creating a new-generation building energy simulation program," *Energy and buildings*, vol. 33, no. 4, pp. 319–331, 2001.

[7] P. A. Mathew, L. N. Dunn, M. D. Sohn, A. Mercado, C. Custudio, and T. Walter, "Big-data for building energy performance: Lessons from assembling a very large national database of building energy use," *Applied Energy*, vol. 140, pp. 85–93, 2015.
[8] H. B. Demuth, M. H. Beale, O. De Jess, and M. T. Hagan, *Neural network design*. Martin Hagan, 2014.
[9] R. Priyadarsini, W. Xuchao, and L. S. Eang, "A study on energy performance of hotel buildings in singapore," *Energy and Buildings*, vol. 41, no. 12, pp. 1319–1324, 2009.
[10] B. Dong, S. E. Lee, and M. H. Sapar, "A holistic utility bill analysis method for baselining whole commercial building energy consumption in singapore," *Energy and buildings*, vol. 37, no. 2, pp. 167–174, 2005.
[11] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling, "Comparison of machine learning methods for estimating energy consumption in buildings," in *Probabilistic Methods Applied to Power Systems (PMAPS), 2014 International Conference on*. IEEE, 2014, pp. 1–6.
[12] R. Kumar, R. Aggarwal, and J. Sharma, "Energy analysis of a building using artificial neural network: A review," *Energy and Buildings*, vol. 65, pp. 352–358, 2013.
[13] J. Ma, J. Qin, T. Salsbury, and P. Xu, "Demand reduction in building energy systems based on economic model predictive control," *Chemical Engineering Science*, vol. 67, no. 1, pp. 92–100, 2012.
[14] P. Radecki and B. Hencey, "Online thermal estimation, control, and self-excitation of buildings," in *52nd IEEE Conference on Decision and Control*. IEEE, 2013, pp. 4802–4807.
[15] J. L. Mathieu, S. Koch, and D. S. Callaway, "State estimation and control of electric loads to manage real-time energy imbalance," *IEEE Transactions on Power Systems*, vol. 28, no. 1, pp. 430–440, 2013.
[16] M. Lundin, S. Andersson, and R. Östin, "Development and validation of a method aimed at estimating building performance parameters," *Energy and Buildings*, vol. 36, no. 9, 2004.
[17] J. Gao and R. Jamidar, "Machine learning applications for data center optimization," *Google White Paper*, 2014.
[18] M. Macas, F. Lauro, F. Moretti, S. Pizzuti, M. Annunziato, A. Fonti, G. Comodi, and A. Giantomassi, "Sensitivity based feature selection for recurrent neural network applied to forecasting of heating gas consumption," in *International Joint Conference SOCO'14-CISIS'14-ICEUTE'14*. Springer, 2014, pp. 259–268.
[19] V. Mnih, H. Larochelle, and G. E. Hinton, "Conditional restricted boltzmann machines for structured output prediction," *arXiv preprint arXiv:1202.3748*, 2012.
[20] G. W. Taylor, G. E. Hinton, and S. T. Roweis, "Two distributed-state models for generating high-dimensional time series," *Journal of Machine Learning Research*, vol. 12, no. Mar, pp. 1025–1068, 2011.
[21] J. Kelly and W. Knottenbelt, "Neural nilm: Deep neural networks applied to energy disaggregation," in *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. ACM, 2015, pp. 55–64.
[22] J. Trindler, R. Zwiker, U. Rutishauser, R. Douglas, and J. Joller, "Discovery of logical structures in a multisensor environment based on sparse events," *8th National Congress of Italian Association for Artificial Intelligence*, pp. 1–11, 2003.
[23] R. Fontugne, J. Ortiz, N. Tremblay, P. Borgnat, P. Flandrin, K. Fukuda, D. Culler, and H. Esaki, "Strip , Bind , and Search : A Method for Identifying Abnormal Energy Consumption in Buildings," *12th international conference on Information processing in sensor networks*, pp. 129–140, 2013.
[24] D. L. Marino, K. Amarasinghe, and M. Manic, "Building energy load forecasting using deep neural networks," in *Industrial Electronics Society, IECON 2016-42nd Annual Conference of the IEEE*. IEEE, 2016, pp. 7046–7051.
[25] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE transactions on neural networks and learning systems*, 2016.
[26] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*. IEEE, 2013, pp. 6645–6649.
[27] D. Neil, M. Pfeiffer, and S.-C. Liu, "Phased LSTM: Accelerating recurrent network training for long or event-based sequences," in *Advances in Neural Information Processing Systems*, 2016, pp. 3882–3890.
[28] "Keras: The Python Deep Learning Library," https://keras.io/, accessed: 2017-08-31.
[29] "Advanced activation layers," https://keras.io/layers/advanced-activations/, accessed: 2017-08-31.
[30] E. Alpaydin, *Introduction to machine learning*. MIT press, 2014.