Deep Learning Techniques for Load Forecasting in Large Commercial Buildings

Cristina Nichiforov, Grigore Stamatescu, Iulia Stamatescu,
Vasile Calofir, Ioana Fagarasan and Sergiu Stelian Iliescu
Department of Automatic Control and Industrial Informatics
University Politehnica of Bucharest
Bucharest, Romania
cristina.nichiforov@aii.pub.ro

***Abstract*—As large scale energy management strategies have gradually shifted the focus from the producer to the consumer side, buildings are starting to play a critical role in the efficient management of the electrical grid. Moreover some buildings have become prosumers by integrating local generation capabilities from renewable sources thus inducing additional complexity into the operation of the energy systems. As alternative to conventional energy consumption modelling techniques, a black-box input-output approach has the ability to capture underlying consumption patterns and trends while making use of the large quantities of data being generated and recorded through dense instrumentation of the buildings. The paper discusses and illustrates an approach to apply deep learning techniques, namely Recurrent Neural Networks implemented by means of Long Short-Term Memory layers, for load forecasting. We focus on large commercial buildings which can be better managed by central operators and where better models can result in significant energy savings and broad economic and social impact. The case study is illustrated on two university buildings from temperate climates over one year of operation using a reference benchmarking dataset for replicable results. The obtained results show promise and can be further used in reliable load management algorithms with limited overhead for periodic adjustments and model retraining.**

***Index Terms*—smart buildings, load forecasting, computational intelligence, long short-term memory**

I. INTRODUCTION

Global urbanisation tendencies have led to significant engineering challenges for the development and management of the built environment. Smart cities are one of the salient examples of a new paradigm that brings together sensing, computing, communication and control to improve the operations of various systems and the well-being of its inhabitants. One of the critical areas of development within a smart city is in the energy sector and the electrical grid, thereby assuring a reliable, clean and cost-effective energy supply to ever increasing urban needs. More specifically our work focuses on large commercial buildings which play a critical role as consumers, prosumers or balancing entities for grid stability. As such, having accurate models that capture underlying patterns and trends driving energy consumption can be used to forecast load profiles and improve high level control strategies. The main objective is to alleviate existing challenges toward grid stability and environmental benefits by leveraging state-of-the-art algorithms.

As building energy use is steadily increasing and has reached almost 40% of primary energy use in many developed countries [1], modern buildings make use of extended instrumentation networks to measure and control many energy related parameters for daily operation. The resulting data streams are handled by information systems such as Building Management Systems (BMS) and stored in local or cloud databases pending further processing or on-line decision making. In existing and older buildings replacing conventional energy meters with smart meters, over wired or wireless communication networks, provides a cost-effective way to collect the relevant data. Based on improved availability and reduced data collection effort for energy measurements time series there are multiple approaches to capture accurate models for energy prediction and control.

As an example, in the recent period, statistical learning methods have seen increased adoption in both research and industry, driven mainly by data availability and exponentially increasing computing resources, including cloud systems. Neural networks are one salient example of statistical learning technique that has shown good results in many types of applications. This is valid for both classification tasks, as well as regression tasks where the objective is to predict an output numerical value of interest. Deep learning techniques build upon the well studied neural network architectures, with increased complexity. This occurs by adding many hidden layers in the overall network as well as many data processing units of various types at each layer of the network. Initially deployed through industry driven initiatives in the areas of multimedia processing and translation systems, other technical applications currently stand to benefit from the availability of open-source algorithms and tools.

The focus of our work is on large commercial buildings as opposed to an equally large market in the individual dwellings and residential sector. This has to do mainly with the economic incentives and return of investment related to energy efficiency

This work has been partially funded by University "Politehnica" of Bucharest, through the "Excellence Research Grants" Program, UPB-GEX. Identifiers: UPB-EXCELENTA-2017 "E-Learning Solutions for Power Engineering", 51/20.09.2017.

projects where small percentage gains on large absolute values of energy use become more attractive to the building operator or owner. The opportunities to finally deploy such intelligent algorithms become greater at scale.

Main contributions of the paper can be thus summarised:

- illustrating a deep learning approach to model large commercial building electrical energy usage as alternative to conventional modelling techniques;
- presenting an experimental case study using the chosen deep learning techniques enabling reliable forecasting of building energy use.

The rest of the paper is structured as follows. Section II discusses related work in the area of energy consumption modelling of direct relevance to our proposed contributions. Section III presents the theoretical background of Recurrent Neural Networks (RNN) implemented with layers of Long Short-Term Memory (LSTM) units and their application for this task. A case study is described in Section IV by applying the deep learning techniques on a reference benchmarking dataset for two large commercial buildings. The main findings are also discussed. Section V concludes the paper and discusses implementation paths of the derived models.

II. RELATED WORK

The application of statistical learning techniques for energy modeling has seen an increased adoption in the recent period. Three factors are identified for this trend:

- better availability of good quality datasets and computational resources that enable extensive testing and validation of the proposed methods;
- commercial and open-source algorithm libraries and software with suitable documentation and examples for a wider audience;
- increased collaboration between algorithm, computing and control experts and domain specialists in the energy and civil engineering fields; this has influenced the design of new deep learning architectures customised mostly for particular applications.

In [2] the authors focus on two deep learning techniques for building energy consumption namely Conditional Restricted Boltzmann Machines (CRBM) and Factored Conditional Restricted Boltzmann Machines (FCRBM). The results are illustrated in comparison to artificial neural networks and support vector machines (SVM) as well as recurrent neural networks. It is shown that for many of the test scenarios: aggregated and submetering data as well as different time horizons and resolution, the investigated methods offer better performance in terms of RMSE on the prediction horizon. The authors of [3] present an end-to-end deep learning approach for load forecasting of commercial buildings by combining stacked autoencoders (SAE) with extreme learning machines (ELM). SAE extracts the relevant features from the input dataset while ELM works as the predictor. The advantages of this particular method are justified in terms of achieving directly the output weights of the networks without iterative backpropagation. Two RNN models based on LSTM units for building energy consumption are evaluated in [4]. An important finding of this study is that RNN methods tend to perform better than others when applied to aggregated energy data and long term dependencies are more difficult to identify. Also the authors use the resulting model to perform missing value imputation on the original time series.

Building on previous own work, in [5] ARIMA and conventional ANN where tested on a locally collected energy consumption dataset. The study of various ANN configurations for load forecasting of buildings was carried out in [6]. The subsequent models can be then integrated in a decision support system as in [7].

III. ELECTRICAL LOAD FORECASTING USING DEEP NEURAL NETWORKS

Due to their extensive utility and good performance, recurrent neural networks (RNN) are becoming a very important tool in situations represented by sequences of "events" with events representing a data point. Recurrent Neural Networks can perform the same computations for all elements in a sequence of inputs and, for example, they can evaluate nonlinear time series problems, such as energy consumption, and provide forecasts. Recurrent Neural Networks have a structure that is different than the frequently applied statistical learning algorithm - artificial neural networks (ANN) and use backpropagation through time (BPTT) [8] or real-time recurrent learning (RTRL) [9] algorithm to compute the gradient descent after each iteration. In other words, RNN is a type of artificial neural network that adds additional weights to the network to create cycles into the network graph in order to control the internal state of the network.

A. Long Short Therm Memory (LSTM)

Long Short-Term Memory neural networks are a particular type of recurrent neural networks. As was mentioned before recurrent neural networks are usually trained using either BPTT or RTRL algorithm, but several researchers showed that training using these methods usually fails because of exploding gradient. LSTM [10] is a recurrent neural network that offers support for time series and sequence data in a network, and help enhance gradient flow over long sequences during training. LSTM addresses the gradient problem through incorporating self-connected "gates" in the hidden units. In a LSTM network the flow of information through the network is handled by reading, writing and removing information from the memory [11], [12].

Figure 1 illustrates the flow of a time series x of length n, ($n \in \mathbb{N}$) through an LSTM layer. In this diagram, h stands for output, also hidden state, and c stands for cell state. The first LSTM hidden unit takes the initial state of the network, (c_0, h_0) and also the first time step of the sequence x_1 and after that computes the first output h_1 and the updated cell state c_1. This process repeats every time step.

The state of the LSTM layer consists of the output state (h) and the cell state (c). On the one hand, the output state

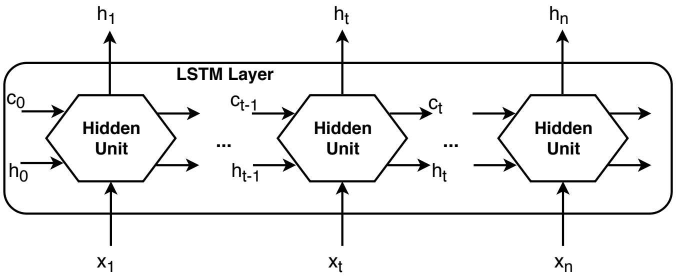


Fig. 1: LSTM layer diagram

at time step t contains the output of the LSTM layer for this time step, and on the other hand the cell state contains information learned from the previous time steps. At each iteration, the layer writes information to the cell state or erases information from it, where the layer controls these updates using ”gates”. The ”gates” represent components that control the cell state and the output state of the layer. There are four such components: the input gate (i) which controls the level of cell state update, the layer input (g) which adds information to cell state, the forget gate (f) which controls the level of cell state reset and the output gate (o) which controls the level of cell state added to output state [12].

The diagram in Figure 2 illustrates the data flow at a specific time step t inside of a hidden unit.

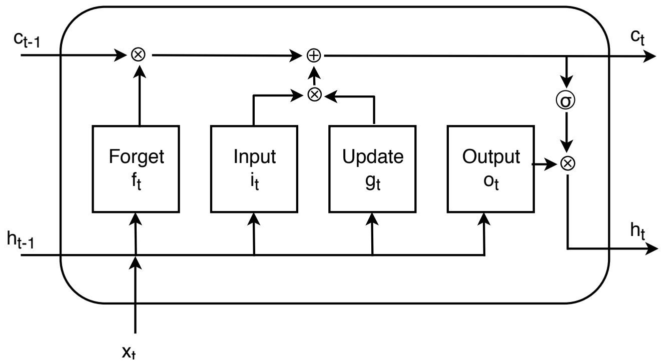


Fig. 2: LSTM hidden unit diagram

A LSTM layer has the following learnable parameters: the input weights (W), the recurrent weights (R) and the bias (b). W, R and b represents matrices that consists of concatenations of the input weights, recurrent weights, and the bias of each component. The structure of each matrix is the following:

$$W = \begin{pmatrix} W_i \\ W_f \\ W_g \\ W_o \end{pmatrix}, R = \begin{pmatrix} R_i \\ R_f \\ R_g \\ R_o \end{pmatrix}, b = \begin{pmatrix} b_i \\ b_f \\ b_g \\ b_o \end{pmatrix},$$

where i, f, g and o represent the input gate, forget gate, layer input and output gate, respectively.

The state of the cell memory at time step t is updated recursively using the following formula [10]:

$$c_t = f_t \otimes c_{t-1} \oplus i_t \otimes g_t, \tag{1}$$

where $\otimes$ stands for Hadamard product and the formulas for every component at time step t are:

$$\begin{aligned} f_t &= \sigma(W_f x_t + R_f h_{t-1} + b_f), \\ i_t &= \sigma(W_i x_t + R_i h_{t-1} + b_i), \\ g_t &= \tanh(W_g x_t + R_g h_{t-1} + b_g), \end{aligned} \tag{2}$$

σ stands for the sigmoid function and $\tanh$ for hyperbolic tangent function.

The output state at time step t is given by the output gate (o) which implements a read function combined with the cell state (c). The process is described by the following formula:

$$h_t = o_t \otimes tanh(c_t), \tag{3}$$

where

$$o_t = \sigma(W_o x_t + R_o h_{t-1} + b_o). \tag{4}$$

B. Benchmarking data sets

The current study presents a LSTM modelling application using different neural network configurations and the assessment of performance between all forecasting LSTM models.

The data sets with active power load used for the experiments are taken from the Building and Urban Data Science (BUDS) Group at the National University of Singapore - `http://www.budslab.org` and are part of a data collection of several non-residential buildings, proposed for performance analysis and algorithm benchmarking [6], [13].

Data was collected every 60 minutes over a 1-year period in two educational buildings with an approximate surface area of 9.000 square meters. The chosen buildings for the study are from university campuses in Chicago (USA) and Zurich (Europe). After the pre-processing of the noise and missing data in the initial data set using the median filter technique two time series data sets were obtained with approximately 8.670 data points each.

Choice of the target buildings was done in conjunction to a local campus building at our university to which a data collection study is currently underway. The determining factors were size - medium to large building, mixed usage pattern - office, laboratory space, some classrooms and non-extreme temperate climate with four distinct seasons [6].

IV. EXPERIMENTAL RESULTS

The input time series for network estimation and testing are illustrated in Figure 3. The upper plot represents the active energy consumed by the university building from Zurich over the reference period and the other one by the university from Chicago.

For this case study we propose various configurations for LSTM neural networks in order to forecast the load in this types of buildings. The goal was to have both an evaluation of the performance metrics of the different structure as well as a computational assessment on the target data set.

The presented work considers the standard LSTM algorithm for load forecasting methodology. All defined networks are standard LSTM neural networks with one sequence input

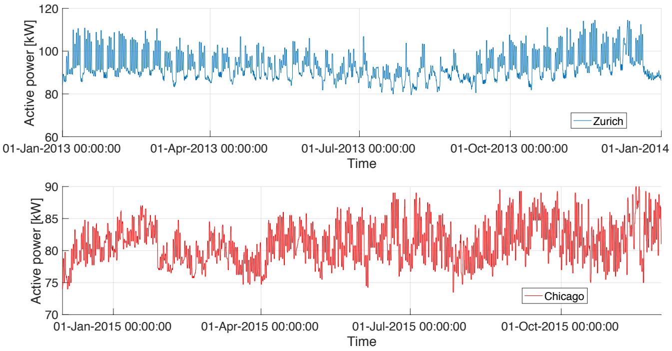


Fig. 3: Data sets used for LSTM estimation and testing

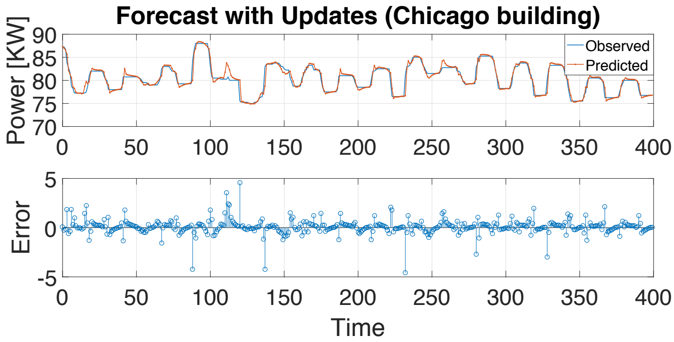


Fig. 4: Prediction result by LSTM neural network with 50 hidden units (Chicago building)

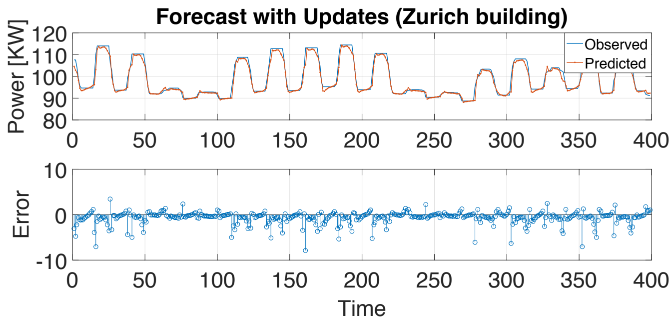


Fig. 5: Prediction result by LSTM neural network with 50 hidden units (Zurich building)

layer, one LSTM layer, one fully connected layer and one regression layer. Each network has a different configuration represented by the number of the hidden units from the LSTM layer. Based on this, the following network structures were implemented:

- *C0:* 5 hidden units in the LSTM layer;
- *C1:* 25 hidden units in the LSTM layer;
- *C2:* 50 hidden units in the LSTM layer;
- *C3:* 100 hidden units in the LSTM layer;
- *C4:* 125 hidden units in the LSTM layer;
- *Z0:* 5 hidden units in the LSTM layer;
- *Z1:* 25 hidden units in the LSTM layer;
- *Z2:* 50 hidden units in the LSTM layer;
- *Z3:* 100 hidden units in the LSTM layer;
- *Z4:* 125 hidden units in the LSTM layer,

where Z stands for the Zurich and C for the Chicago.

Regarding the training process, the Adaptive Moment Estimation (ADAM) algorithm was used. ADAM is an algorithm for first order gradient based optimisation of stochastic objective functions with momentum. The algorithm computes learning rates that can adapt in a automatic manner to the loss function which is optimised, for each parameter from estimates of first and second moments of the gradients. It maintains an element-wise moving average of the parameter gradients and their squared values, respectively [14].

In the literature is demonstrated that the algorithm is efficient in terms of computational time, has little memory requirements and is suitable for large data problems.

Regarding the learning rate, after few tests, it was concluded that the right solution is the following: the initial learning rate was set to 0.1 and after that, the learning rate was reduced by a factor of 0.2 every 100 epochs. Also, the properly maximum number of epochs for training was chosen to be 200.

Figure 4 and 5 presents the prediction response by the LSTM neural network with 50 hidden units in the LSTM layer versus real data for Chicago building and Zurich building, respectively. The plots demonstrate that the forecasting performances of the LSTM models for the testing data sets is very good.

To evaluate the prediction models, three performance metrics were used: Mean Squared Error (MSE), Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE). The metrics are listed in the following equations:

$$\begin{aligned} MSE &= \sum_{1}^{n} \frac{(Y_t - Yp_t)^2}{n} \\ RMSE &= \sqrt{\sum_{1}^{n} \frac{(Y_t - Yp_t)^2}{n}} \\ MAPE &= \frac{1}{n}\sum_{1}^{n} \left| \frac{Y_t - Yp_t}{Y_t} \right| 100 \end{aligned} \tag{5}$$

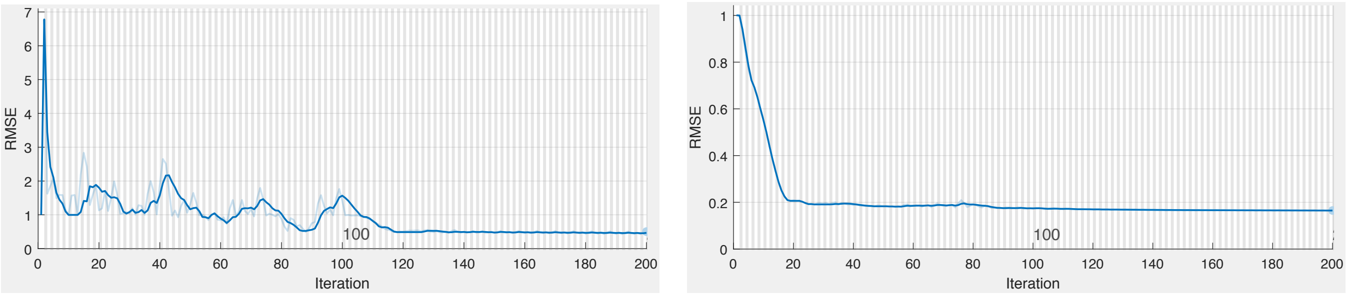
where n represents the number of samples, Y_t and Yp_t stand for the actual data and predicted data, respectively.

Table I and Table II show the computational time, MSE, RMSE and MAPE errors on testing data sets for different number of hidden units in the LSTM layer.

TABLE I: Forecasting performance of each LSTM neural network (Chicago building)

	C0	C1	C2	C3	C4
Time(s)	75	93	143	247	330
MSE	0.6295	0.6132	**0.5553**	0.7486	0.9555
RMSE	0.7934	0.7831	**0.7452**	0.8652	0.9775
MAPE(%)	0.5623	0.5091	**0.4945**	0.5535	0.8177

From the testing performance point of view, as shown in Table I and II, the smallest prediction errors and best precision for both the Chicago building data and Zurich building data,



(a) (b)

Fig. 6: Training Performance a) Worst b) Best

TABLE II: Forecasting performance of each LSTM neural network (Zurich building)

	Z0	Z1	Z2	Z3	Z4
Time(s)	76	94	150	275	355
MSE	2.3846	2.1732	**2.0506**	2.2506	20.34
RMSE	1.5442	1.4742	**1.432**	1.5002	4.5107
MAPE(%)	0.9197	0.9008	**0.828**	0.9958	3.4684

appear for the configuration with 50 hidden units in the LSTM layer. The computation time is relevant for the test computer with an 3.4 GHz i7 quad core processor and 16GB RAM and the algorithms running under MATLAB R2018a software environment from MathWorks, which is widely used by engineers and scientists in industry and academia for a range of applications, including deep learning and machine learning, signal processing and communications.

Figure 6 illustrates the evolution of the RMSE metric over the training horizon of 200 epochs for the best and worst case scenarios. The outcome is illustrated on the Zurich dataset where the best results are yielded in the Z2 - 50 units in the hidden layer. The worst training results have been achieved in the Z4 - 125 units in the hidden layer configuration and it can be seen how the RMSE bounces back and forward over the training horizon due to overfitting of the dataset by the complex network structure. The best algorithm converges to a minimum test RMSE value in under 100 iterations while for the worst case convergence to a higher test RMSE minimum value is achieved in around 120 iterations.

V. Conclusions

The paper presented an approach to apply deep learning techniques to the problem of energy consumption prediction of large commercial buildings. The technique of choice uses recurrent neural networks with a layer of LSTM units of varying dimension. The presented case study is replicable given the usage of open benchmarking datasets on two reference university buildings. We investigate the results in terms of network complexity and identify a suitable network structure that achieves the best accuracy on the input data sets while avoiding overfitting the data. The models that were implemented and evaluated are suitable for online optimisation provided with the right data streaming and computing infrastructure.

As there are many types of models that can be applied depending on the data and the underlying processes, building characteristics and consumption patterns, the final choice can also be guided through domain expertise. Further research is currently underway to use one the derived black-box models within predictive control algorithms that allow the implementation of load management strategies e.g. by modulating chiller output power in conjunction to high energy prices, weather variations or other unexpected events. Also on-site energy storage can bring significant benefits to the energy management strategy with increased problem complexity.

References

[1] U. Berardi, “Building energy consumption in us, eu, and bric countries,” *Procedia Engineering*, vol. 118, pp. 128 – 136, 2015, defining the future of sustainability and resilience in design, engineering and construction. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877705815020664

[2] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling, “Deep learning for estimating building energy consumption,” *Sustainable Energy, Grids and Networks*, vol. 6, pp. 91 – 99, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2352467716000163

[3] C. Li, Z. Ding, D. Zhao, J. Yi, and G. Zhang, “Building energy consumption prediction: An extreme deep learning approach,” *Energies*, vol. 10, no. 10, 2017. [Online]. Available: http://www.mdpi.com/1996-1073/10/10/1525

[4] A. Rahman, V. Srikumar, and A. D. Smith, “Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks,” *Applied Energy*, vol. 212, pp. 372 – 385, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306261917317658

[5] C. Nichiforov, I. Stamatescu, I. Făgărăşan, and G. Stamatescu, “Energy consumption forecasting using arima and neural network models,” in *2017 5th International Symposium on Electrical and Electronics Engineering (ISEEE)*, Oct 2017, pp. 1–4.

[6] C. Nichiforov, G. Stamatescu, I. Stamatescu, I. Fagarasan, and S. S. Iliescu, “Intelligent load forecasting for building energy management,” in *2018 14th IEEE International Conference on Control and Automation (ICCA)*, Jun 2018.

[7] I. Stamatescu, N. Arghira, I. Fagarasan, G. Stamatescu, S. S. Iliescu, and V. Calofir, “Decision support system for a low voltage renewable energy system,” *Energies*, vol. 10, no. 1, 2017. [Online]. Available: http://www.mdpi.com/1996-1073/10/1/118

[8] P. J. Werbos, “Backpropagation through time: what it does and how to do it,” *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, Oct 1990.

[9] R. J. Williams and D. Zipser, “Backpropagation,” Y. Chauvin and D. E. Rumelhart, Eds. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1995, ch. Gradient-based Learning Algorithms for Recurrent Networks and Their Computational Complexity, pp. 433–486. [Online]. Available: http://dl.acm.org/citation.cfm?id=201784.201801

[10] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.8.1735
[11] D. L. Marino, K. Amarasinghe, and M. Manic, “Building energy load forecasting using deep neural networks,” *CoRR*, vol. abs/1610.09460, 2016. [Online]. Available: http://arxiv.org/abs/1610.09460
[12] S. Srivastava and S. Lessmann, “A comparative study of lstm neural networks in forecasting day-ahead global horizontal irradiance with satellite data,” *Solar Energy*, vol. 162, pp. 232 – 247, 2018.
[13] C. Miller and F. Meggers, “The building data genome project: An open, public data set from non-residential building electrical meters,” *Energy Procedia*, vol. 122, pp. 439 – 444, 2017.
[14] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980