A neural network approach to predicting urban building energy consumption

Rohan Aras, Alex Nutkiewicz, Max O'Krepki
{rohana, alexer, maxo}@stanford.edu

Abstract
As the world rapidly urbanizes, the built environment will become increasingly responsible for the world's primary energy usage and greenhouse gas emissions. However, designers and engineers are unable to predict the energy performance of these buildings at a high accuracy because of their inability to account for the intra-building energy dynamics and interdependencies that can have a substantial impact on building energy use. This paper tests two methods: a multilayer perceptron (MLP) and residual network (ResNet) to learn the hidden context affecting building energy consumption and predict at a high accuracy. *The results show that a Deep Residual Network with 2 channels in each block and 128 output channels outperforms other algorithms and shows a 74.9% prediction improvement over traditional energy simulation approaches.*

1 Introduction

Cities account for over 75% of all primary energy usage and over 80% of greenhouse gas emissions, with the largest portion of such consumption (more than 40%) and emissions coming from the built environment [1]. Because 90% of urban buildings are estimated to be energy inefficient and up to 30% of a building's energy consumption is wasted, buildings represent a tremendous opportunity to enhance the sustainability of cities [2].

Building energy modeling (BEM) is a tool widely used in building design and construction to understand and predict the energy performance of buildings. However, accurate prediction of building energy consumption remains a challenge because these BEMs often only model a single building and do not account for the building's urban context - including both surrounding buildings and the urban microclimate. Additionally, the time and resources needed to create a building energy model for hundreds or thousands of buildings across a city requires an infeasible amount of time and resources to execute efficiently. And while there are energy forecasting models that use metered energy data as both inputs and outputs, these statistical methods cannot accurately estimate the energy implications of changes made to buildings during their lifecycle (e.g., energy efficiency upgrades, demand-side management) [3]. This project aims to integrate deep learning into an urban building energy simulation to more accurately predict the energy consumption of urban buildings [Figure 1]. *The inputs to our algorithm consist of fifteen-minute interval energy data created through a building energy model as well as weather data from the location of study. We then test a multilayer perceptron and residual network to output a predicted time series of energy consumption for the same buildings.*

2 Related Work

The advent of smart meters and open data initiatives in major cities has made energy consumption data available for prediction models [3]. However, seldom is this data used in the creation of even "state-of-the-art" building energy models, which instead rely on physics-based equations to predict building energy consumption for each year [4]. Because there is no existing hybrid model that integrates machine learning and energy simulation, we look to other examples in urban systems with similar spatial-temporal characteristics.

Current machine learning-based algorithms for predicting building energy consumption include ordinary least squares, which are highly susceptible to outlier and are generally unable to model the nonlinearities in building energy consumption [5], support vector regression [6]; and clustering, while computationally efficient, is more commonly used to perform classification on a specific category of buildings and is rarely conducted on an urban level [7]. Lately, researchers in this field have begun to use neural networks as they are able to learn complex relationships in a multi-dimensional domain and are able to model noisy data produced by building energy systems [8]. Convolutional neural networks (CNNs) are particularly popular for this type of prediction problem because they can represent data, specifically time series data, in a grid topology and have shown to be as effective as other complex deep learning methods [9].

Looking to other non-energy related examples, [10] also utilizes a CNN but in the context of transportation by creating features representing as relationships between roads and traffic speed, organized by time and space. The resulting matrix is viewed as a channel of an image similar to how CNNs interpret images by pixels and color. Taking inspiration from this particularly novel approach to modeling a spatial-temporal environment, we sought out to explore this further in our study.

3 Dataset and Feature Engineering

In order to create a more accurate, robust, and scalable urban energy simulation, we tested two types of neural networks: Multi-layer Perceptrons (MLP) and Residual networks (ResNets). We theorize that the primary source of uncertainty in building energy modeling arises from the "hidden" urban context impacts that are not captured currently by traditional BEM. By modeling measured individual building energy consumption as a function of the simulated energy consumption for individual buildings, we hope that this model will better account for these hidden urban scale interdependencies. In the case of neural networks, we hypothesize these relationships will be captured in the hidden layer(s). This can be seen in Figure 1.

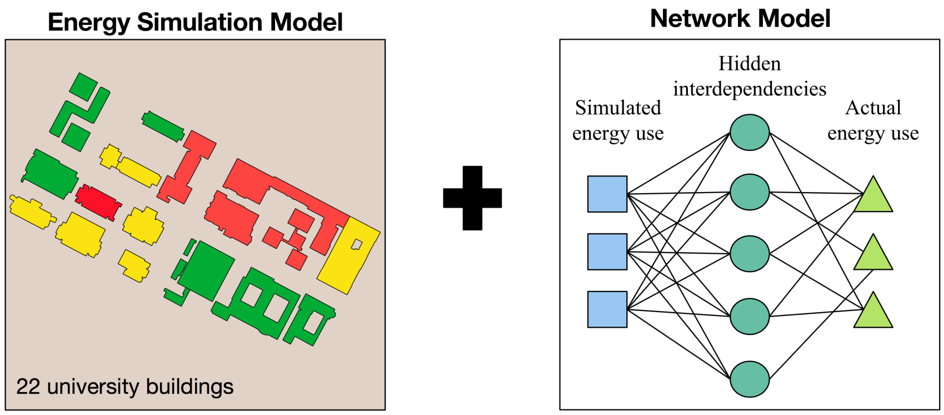

Figure 1. Simplified workflow for predicting building energy consumption. Inputs to the neural network include simulated energy usage for all buildings in the study area, and outputs are the measured energy usage for each building – both at 15-minute time intervals. The hidden layers represent the urban context influencing overall energy consumption.

To gather the data needed for the network algorithms, we first created building energy models for 22 densely co-located buildings on a university campus. Each BEM has a time series output of 15-minute intervals for energy usage that will serve as the inputs to the neural network. For the network's output, we acquired 15-minute interval energy consumption for the 22 buildings spanning two years (~70,080 observations). Because weather has a direct correlation with building energy consumption [9], hourly weather data was collected for the local area of study as additional features to be used in the prediction [11]. The selected weather features include solar radiation, outdoor dry-bulb temperature, and relative humidity.

After generating all 22 BEMs and collecting the interval time series data for each building in the study area, the data was cleansed for missing or outlier values. The hourly weather data was expanded to fifteen-minute intervals with the assumption that the weather did not change during the hour. Missing metered energy values were imputed using the R mice package. To create the MLP, we add 3 features from our weather dataset (*solar radiation*, *outdoor dry-bulb temperature*, and *relative humidity*) to give the input space a total of 25 input features and 22 outputs. For the ResNet, we one hot encode *day of the week* and *month* into our input space, increasing the number of input features to 41 while keeping the number of output features at 22.

4 Methods/Network Architectures

4.1 Multi-layer perceptron (MLP)

Our baseline of comparison is a basic MLP that consists of three fully connected layers: one input, one hidden, and one output. Each layer consists of 64 neurons and is designed with a ReLU nonlinearity activation function. A basic layer in the MLP is formalized as:

$$y = \boldsymbol{W} \cdot \hat{\boldsymbol{x}} + \mathbf{b}$$
$$h = ReLU(y)$$

We chose a ReLU activation function because it generally has the ability to help networks stack deeper. The number of neurons and layers were chosen through experimentation. The results for each of our models can be seen in Section 5.

4.2 Deep residual networks (ResNets)

New research in urban systems looks towards CNNs as a method of assessing spatial-temporal issues as their characteristic locally connected convolutional layers enable them to efficiently deal with spatially correlated problems [9]. However, we were interested in exploring ResNets because they apply convolutions in a similar manner but do not experience the high training losses associated with degradation [14, 16]. Deep residual networks (ResNets) are multi-layer neural networks in which each layer consists of a residual module f_i and an identity mapping connection that can bypass f_i [15], [16]. Because layers in a residual network can consist of multiple convolutional layers, they are often referred to instead as "residual blocks." Residual learning takes the form of

$$H(x) = x + F(x)$$

Where *H(x)* is the underlying mapping, *F(x)*, the residual block, is a series of convolution layers with batch normalization [17] and a leaky ReLU activation function, and *x* is the original input space. It can be further represented as

$$F(x) = \sigma(W_2\sigma(W_1x) + x)$$

Where W_1 and W_2 represent the weights for each convolutional layer in the residual block and σ is a leaky ReLU activation function. In the case where the residual block poorly estimates the underlying mapping, its weight (W) can be driven to zero and the layer can instead approximate the identity mapping with the original input space. The specific architecture of this function may vary depending on application, but this study has chosen the design based on the results of [15] and can be seen in Figure 2.

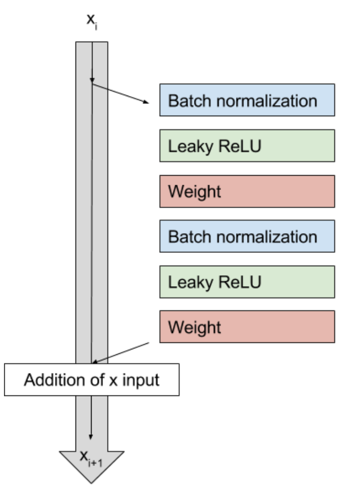

Figure 2. Pre-activation function used for ResNet architecture.

For our implementation, we shape our input data into a 730 signals of 96 time steps by 41 channels (buildings and one-hot vectors) [Figure 3]. This input is fed into a 1D convolutional layer that outputs 64 channels, where the output is passed through K residual blocks, each containing two 1D convolutions, pre-activated as explained in Figure 2. The output of these blocks is passed through a final 1D convolution that outputs 22 channels, where it is added back to the building channels of the original inputs for the models final result.

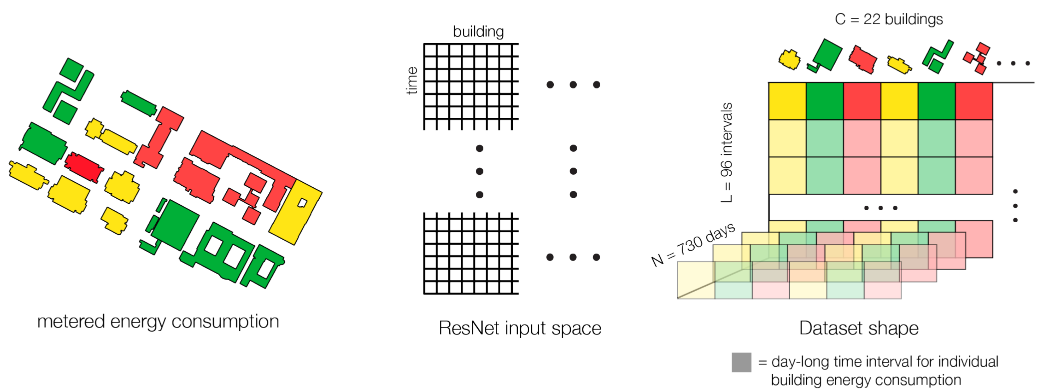

Figure 3. ResNet dataset shape

5 Results & Discussion

In order to select the best model, we compared the error rates of prediction to traditional error rates used in traditional building energy modeling [12]. These error rates are based on ASHRAE Guideline 14-2002, which are based on energy use intensity (EUI), which is the sum of gas and electricity consumption (kWh) per square meter of floor area [13]. We evaluate accuracy based on Mean Squared Error (Equation 2).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Equation 2. MSE used to evaluate all models.

Our results show that the residual network significantly outperformed each of the tested MLP models (Table 1). However, while the ResNet with 8 channels/block and 128 output channels performs with the best train error, it breaks down using the dev set is not longer able to perform as well. As the number of layers in each residual block increases, the model starts to become unstable and therefore increases the dev MSE. This is likely because the input data is needed often to estimate the underlying mapping, so if there are more layers in each block, this cannot happen as frequently. Compared to the baseline model that only relies on simulation data to predict building energy consumption, we see a 74.7% improvement in prediction ability between the top performing model (ResNet with 2 blocks and 128 output channels) and the baseline simulation approach.

We chose to evaluate our models over 30 epochs because we noticed that adding any extra resulted in their results becoming more and more unstable. The learning rate used for all models was 0.0001, which was also determined through experimentation. We evaluated MLPs using a higher learning rate, but did not notice any improvement on the prediction ability.

Finally, we want to point out the results for the multilayer perceptron because of the difference between the training and dev MSE. As shown in Table 1, the dev MSE is lower for both MLP models. We hypothesize this is the result of the imputation we did on the data prior to running the models, although further experimentation should be done to understand the concrete reason for this phenomenon.

Table 1. Quantitative results for evaluated neural networks. Test MSE was evaluated using the best performing dev set model (ResNet with 2 blocks and 128 output channels).

Model Type, Selected Hyperparameters	**Num. of Observations**	**Train MSE**	**Dev MSE**	**Test MSE**
Baseline: Simulation data, with no machine learning	**N/A**	**97.6788**	**102.262**	**108.3428**
MLP, 64 neurons, 1 hidden layer	70,080	67.6473	65.0505	
MLP, 64 neurons, 3 hidden layers	70,080	61.1613	59.4322	
ResNet, 2 channels/block, 64 output channels	70,080	24.0158	27.5494	
ResNet, 4 channels /block, 64 output channels	70,080	22.1044	24.8864	
ResNet, 8 channels /block, 64 output channels	70,080	20.9799	36.6208	
Top Performer: ResNet, 2 channels **/block, 128 output channels**	**70,080**	**21.6470**	**24.8464**	**27.3607**
ResNet, 4 channels /block, 128 output channels	70,080	20.5877	36.6114	
ResNet, 8 channels /block, 128 output channels	70,080	19.7524	105.778	
Average error, ResNet	N/A	21.5142	42.7154	

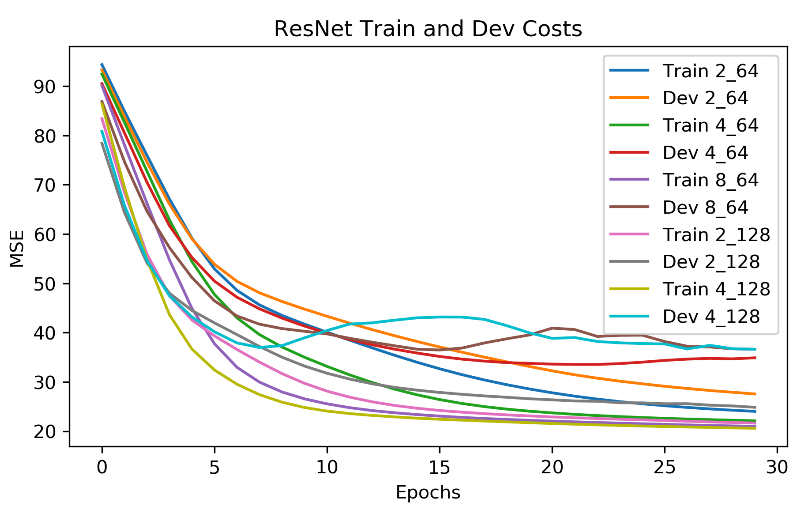

Figure 4. ResNet train and dev set results. These results show that models with more output channels tend to break down as the number of epochs increases.

6 Conclusion & Future Work

As urban density continues to increase, it is imperative that the buildings in which people reside, work, and interact become more energy efficient. Our results show that Deep Residual Networks, specifically with 2 convolutions per block and 128 output channels, were the most effective in estimating building energy use. We hypothesize that by modeling the network using defined signals of time series data we were able to find more of the patterns reflective of building energy consumption. Additionally, because there are so many spatially-related factors that influence energy use, utilizing the locally connected convolutions allowed more a more accurate end result.

Future work on this project will aim to further reduce the MSE for the Residual Network and to evaluate it against other state-of-the-art urban simulation-based approaches for predicting energy consumption. We would like to evaluate these errors at different time intervals (e.g., daily, monthly, yearly) and spatial scales (e.g., individual building scale, city-wide scale) to further understand how a hybrid simulation and deep learning approach to energy prediction could improve the process of building design and energy efficiency retrofits.

7 Contributions

For this project, Rohan was responsible for implementing the ResNet models. Alex led the effort on literature review, data collection, cleaning, and poster preparation. And finally, Max led preparation and analysis of the MLP models.

8 References

[1] United States Energy Information Administration, "Annual Energy Outlook 2017," 2017. [Online]. Available: https://www.eia.gov/outlooks/aeo/.

[2] Breaking Energy, "The Top Ten Ways We Waste Energy in Water and Buildings," 2011. [Online]. Available: http://breakingenergy.com/2011/07/26/the-top-ten-ways-we-waste-energy-and-water-in-buildings/.

[3] Z. Yang and B. Becerik-Gerber, "A model calibration framework for simultaneous multi-level building energy simulation," *Appl. Energy*, vol. 149, pp. 415–431, 2015.

[4] C. Cerezo Davila, C. F. Reinhart, and J. L. Bemis, "Modeling Boston: A workflow for the efficient generation and maintenance of urban building energy models from existing geospatial datasets," *Energy*, vol. 117, pp. 237–250, 2016.

[5] N. El-Gohary, "Data-driven Residential Building Energy Consumption Predicition for Supporting Multiscale Sustainability Assessment," *Proc. Int. Work. Comput. Civ. Eng. 2017*, p. 9, 2017.

[6] R. K. Jain, K. M. Smith, P. J. Culligan, and J. E. Taylor, "Forecasting energy consumption of multi-family residential buildings using support vector regression: Investigating the impact of temporal and spatial monitoring granularity on performance accuracy," *Appl. Energy*, vol. 123, no. Supplement C, pp. 168–178, 2014.

[7] D. Hsu, "Characterizing Energy Use in New York City Commercial and Multifamily Buildings," in *ACEEE summer study on energy efficiency in buildings*, 2012.

[8] M. W. Ahmad, M. Mourshed, and Y. Rezgui, "Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption," *Energy Build.*, vol. 147, no. Supplement C, pp. 77–89, 2017.

[9] K. Amarasinghe, D. L. Marino, and M. Manic, "Deep neural networks for energy load forecasting," *2017 IEEE 26th Int. Symp. Ind. Electron.*, pp. 1483–1488, 2017.

[10] Z. Dai, Z. He, and Y. Wang, "Learning Traffic as Images : A Deep Convolution Neural Network for Large - scale Transportation Network Speed Prediction," *Sensors MDPI*, pp. 1–16, 2017.

[11] National Renewable Energy Laboratory, "Weather Data," 2017. [Online]. Available: https://energyplus.net/weather.

[12] J. Sokol, C. Cerezo Davila, and C. F. Reinhart, "Validation of a Bayesian-based method for defining residential archetypes in urban building energy models," *Energy Build.*, vol. 134, pp. 11–24, 2017.

[13] J. M. Calm *et al.*, "ASHRAE STANDARD Energy Standard for Buildings Except Low-Rise Residential Buildings," *Society*, vol. 8400, pp. 404–636, 2007.

[14] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2017–May, pp. 1578–1585, 2017.

[15] S. Wu, S. Zhong, and Y. Liu, "Deep residual learning for image recognition," *Multimed. Tools Appl.*, pp. 1–17, 2017.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," *Lect. Notes Comput. Sci.*

(including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9908 LNCS, pp. 630–645, 2016.
[17] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” 2015.