Energy Model Machine (EMM)

Instant Building Energy Prediction using Machine Learning

Mohammad Rahmani Asl[1] , Subhajit Das[2] , Barry Tsai[3] , Ian Molloy[4] , Anthony Hauck[5]
[1,3,4,5] Autodesk Inc. [2] Georgia Institute of Technology
[1,3,4,5] {mohammad.asl|barry.tsai|ian.molloy|athony.hauck}@autodesk.com
[2] das@gatech.edu

In the process of building design, energy performance is often simulated using physical principles of thermodynamics and energy behaviour using elaborate simulation tools. However, energy simulation is computationally expensive and time consuming process. These drawbacks limit opportunities for design space exploration and prevent interactive design which results in environmentally inefficient buildings. In this paper we propose Energy Model Machine (EMM) as a general and flexible approximation model for instant energy performance prediction using machine learning (ML) algorithms to facilitate design space exploration in building design process. EMM can easily be added to design tools and provide instant feedback for real-time design iterations. To demonstrate its applicability, EMM is used to estimate energy performance of a medium size office building during the design space exploration in widely used parametrically design tool as a case study. The results of this study support the feasibility of using machine learning approaches to estimate energy performance for design exploration and optimization workflows to achieve high performance buildings.

Keywords: *Machine Learning, Artificial Neural Networks, Boosted Decision Tree, Building Energy Performance, Parametric Modeling and Design, Building Performance Optimization*

INTRODUCTION

The building sector as the largest consumer of the United States primary energy has been seeking to take necessary actions to reduce its energy use. Due to the considerable impact of the buildings on the environment and with the rise in environmental concerns, designers are increasingly expected to conform appropriate minimum requirements regarding energy efficiency (Rahmani Asl et al. 2015). They need to identify design parameters with the significant impact on the building energy performance and optimize them in the process of design to achieve building models with higher energy efficiency (Yu et al. 2010).

Multiple software applications have been developed for simulating building energy performance, renewable energy, and sustainability in buildings (Zhao and Magoulès 2012). While these simulation tools produce high accuracy results, the simulation process is computationally expensive and time consuming. On the other hand, these tools are based on physical principles and they require high levels of expertise and detail building and environmental parameters as input. As a result, building energy performance analysis is typically performed for final design validation and at the later phases of design process.

During the early design stages, designers often need to quickly explore multiple design alternatives and optimize multiple performance factors at the same time make preliminary decisions. At this phase, designers do not require high fidelity simulations for decision making and they just need to compare multiple design options and find the most appropriate alternatives for their problem. This process is difficult with whole building energy simulation tools due to slow feedback from conventional energy simulation engines (Tsanas and Xifara 2012). Therefore, building science researchers proposed different approaches for developing practical surrogate models to replace actual simulation in the early stages of design. Significant efforts have been done to make conceptual design tools environments interactive, so that designers can get instant feedback for continuous design iterations. One of the alternatives to high accuracy energy simulation is the use of fast surrogate models (Guo et al. 2016). Among these models, data-driven surrogate models become more and more practical and important because getting access to large volume of simulation data and computation power is getting easier. Over the past years, machine learning approaches and in specific deep learning methods were very successful in learning from data.

In this paper we introduce Energy Model Machine (EMM), a Machine Learning (ML) based tool that uses Artificial Neural Networks (ANNs) and Boosted Decision Tree (BDT) methods trained on the existing simulation results to predict the energy performance of the buildings without the need to perform the actual simulation. EMM is developed to provide instant feedback to designers in the early phases of building design and guide them to better building energy and environmental performance. EMM uses a data set of simulated models for optimising the weight parameters of ANNs algorithm and and BDT varies maximum depth of pruning and number of models for the ensemble set for each experiments. The trained models is then used to predict the energy performance of the newly submitted models by users and provide instant feedback to help them design energy efficient buildings. Basically EMM is an Artificial Intelligence that uses the results of the existing simulations to predict the performance of the user models as a service. It can be integrated with generative design and parametric performance analysis workflows to enable designers study a large number of design alternatives in a short period. In this paper, we provide details about the EMM development process, and we compare the EMM predicted results of arbitrary models and actual building energy simulation results. The paper provides a case study of use of EMM in a generative design application to explore the design space of a medium size office building considering annual energy use as the main performance factor.

RELATED WORK

Building energy simulation engines are widely used for energy performance prediction to help designers in the process of high performance building design since practice has shown that these tools can often generate results which accurately reflect actual building energy use (Tsanas and Xifara 2012). These tools use physical rules and principles to calculate thermal dynamics and building energy use. Most of the initial work on developing building energy simulation algorithms was done a few decades ago. Nevertheless, these tools became more accessible to designers over the past few years with the advancement of computational services. The U.S. Department of Energy (2012) has been publishing the “Building Energy Software Tools Directory” that provides detail information for over four hundred simulation tools for evaluating energy efficiency, renewable energy, and sustainability in buildings. Crawley et al. (2008) provided a report comparing the features and capabilities of twenty major building energy simulation applications. These resources can be used as a reference for detail information about the widely used building energy simulation tools.

Using advanced building energy simulation tools may provide reliable solutions to estimate the energy performance of building design alternatives; however this process can be very time-consuming and requires user-expertise in a particular program. Hence, in practice designers have to rely on surrogate models to study the energy performance of building design alternatives in generative design and optimization workflows to explore building design space and optimize building performance. In the literature there are various studies that used surrogate models to predict building energy performance. In multiple studies, regression model is used to correlate one or multiple building parameters to building energy consumption (Bauer 2008; Ansari et al. 2005; Catalina et al. 2008; Lam et al. 2010; Al Gharably et al. 2016). Other machine learning methods such as Support Vector Machine (SVM) (Dong et al. 2005; Lee et al. 2009), Decision Tree (Yu et al. 2010), and Artificial Neural Network (ANNs)(Kalogirou et al. 1997; Ben-Nakhi and Mahmoud 2004; Ekici and Aksoy 2009; Zhang et al. 2010) has been used to predict building energy performance. In this study we chose ANNs they are the most widely used artificial intelligence models in the application of building energy prediction they are good for solving complex problems. EMM also uses BDT as it is able to generate predictive models with interpretable flowcharts that enable users to quickly understand the model.

ENERGY MODEL MACHINE (EMM)

Building Energy Model Representation

In this study, we use building design parameters to represent building energy model. These parameters are identified based on their architectural and functional relevance and their potential impact on building energy use. These parameters can be categorised into three groups: 1) geometry parameters, 2) construction parameters and 3) load parameters. Table 1 lists all of the parameters studied for EMM.

Table 1
List of parameters studied in Energy Model Machine

Number	Parameter
1	Climate Zone
2	Number of Floors
3	Total Exterior Wall Area (and Wall Area Height) *
4	Wall Weighted U-Value
5	Wall Area (North, South, East, West) (and Wall Area Height)
6	Total Window Area (and Window Area Height)
7	Window Area (North, South, East, West) (and Window Area Height)
8	Window to Wall Ratio (North, South, East, West)
9	Weighted Window U-Value (North, South, East, West)
10	Weighted Window Solar Heat Gain Coefficient (North, South, East, West)
11	Total Roof Area (and Roof Area Height)
12	Weighted Roof U-Value
13	Total Raised Floor Area (and Floor Area Height)
14	Total Interior Floor Area (and Floor Area Height)
15	Total Slab on Grade Area
16	Shade Area (North, South, East, West)
17	Weighted Lighting Power Density
18	Weighted Plug Load Efficiency
19	Weighted Infiltration
20	HVAC Type
21	Total Envelope Area / Volume
22	Total Floor Area / Volume
23	Total Envelope Area / Total Floor Area

* Area Height is the area multiply by height of each surface and it is used to track the geometry characterizations that are not tracked by other parameters

In this study, we considered building wall, roof, and floor area and window area by direction. Furthermore, we studied the impact of cross-terms parameters that consider wall, window, roof, floor area with their level height in the building as potential variables. These parameters are designed as a set of new geometry related parameters that help to track the variations in geometry more accurately and they are not studied in the previous studies focused on ML-based building energy performance prediction. Also, we gathered directional resolution for walls and windows to increase the accuracy of the regression models. Parameters such as U-Value and Solar Heat Gain Coefficient (SHGC) of building objects are included in the model as construction parameters. Lighting Power Density (LPD), Equipment Power Density (EPD), and Infiltration are some of the load parameters that are included in the model.

Building energy models are created using Autodesk Insight energy analytical model creator from architectural (conceptual and detailed) models and exported as Green Building XML (gbXML) files. We prepared the input parameters for machine learning algorithm by running a parser on gbXML files for the models in the data set and prepare a data set of desired parameters. The ML algorithm is then run on this data set. The following section describes the underlying working principles of some of the ML algorithms implemented on the data set.

Machine Learning

We used two machine learning methods, Artificial Neural Networks (ANNs) and Boosted Decision Tree (BDT), for EMM. These two methods have been successfully used to predict building energy performance in other studies. ANNs are very good in modeling complex relationships between input parameters and outputs to predict accurate results. However, ANNs usually operate as black box and are very difficult to interpret especially for architects and designers in case they want to understand more about the generated model and the way it works. EMM also uses BDT method which is much simpler to utilize andits result can be interpreted easily compared to ANNs.

Data. Our dataset comprises of over 180,000 data points, each representing a building energy simulation output. The Data has over 97 features including both categorical and quantitative data types. For our experiment and prototype we stored the data as a local csv file for data transformation, cleaning and building our machine learning model. However, for future use cases with more data points, we envision to setup a database and use the current software architecture on it. We cleaned the data of any unwanted features like, project ids, weather id, or sparse fields which did not add much to the information set any way. After removing such features from the data, we were left with 87 features.

Data Modeling. We built our machine learning models with an iterative approach. On each iteration of model building we verified their performance by 10 folds' cross validation on the whole data set. With multiple iterations, we compared Root Mean Square values and R2 Scores across different instances obtained by tweaking model parameters. We picked the model instance with best performance based on the above metrics. Next we split the dataset into 67% training set and 33% test sets. We trained our selected model instance on the training set. Our case, is a regression problem, where out of the 87 features, we have 86 independent variables ('X'). From our dataset, our dependent variable or unknown output 'Y', is the Energy Simulation Output. We computed our model performance based on how accurate the prediction of the energy simulation output is with respect to correct results as present in the test data set. We conducted experiments on two machine learning models ANNs and BDT. We also tested the results with and without feature selection or dimensionality reduction algorithms. Our intent was to explore the trade-off between speed gain and accuracy from the various experiments.

Technology Stack. We used Python for all the data cleaning, data transformation and model building processes, using numpy, pandas, and scipy packages (Walt et al. 2011). For machine learning models, we used scikit learn (Pedregosa et al. 2011) and Keras deep learning toolkit using Theano as a backend to handle the computational heavy lifting for ANNs. For data visualisation to evaluate model performance and to compare model output, we used Python's Matplot (Hunter 2007) library package. Below we describe our machine learning models and experiment setup in detail.

Artificial Neural Networks (ANNs)

ANNs are the most widely used artificial intelligence models in the application of building energy prediction since it is good for solving nonlinear problems with complex interdependencies. In the past twenty years, researchers have applied ANNs to analyze various types of building energy consumption in a variety of conditions, such as heating/cooling load, electricity consumption, sub-level components operation, and estimation of usage parameters. For more information please refer to Krarti (2003) and Dounis (2010) research which provide review studies of artificial intelligence methods in the application of building energy systems.

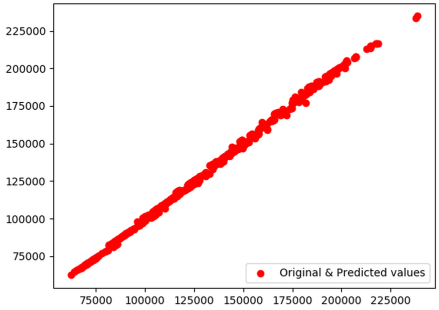
Interest in this direction resurfaced in, 1982, when John Hopfield (Hopfield 1982) in his paper presented methods to create useful systems, using bidirectional connections between neurons. The idea was to learn from biological processes and mimic them on a computer system to help model systems better. With the current advent of Big Data sources and commendable advances in hardware and compute speed availability, use of these systems, has potential to show very good results. Neural networks are capable of generating their implicit rules by learning from provided instances. Their ability to generalise the model has been proved superior to other comparable machine learning systems spanning application in a wide array of fields (Ayat and Pour 2014).

ANNs - Experimental Setup. We created ANN models using Keras library with Scikit Learn package in python. We followed the same pipeline of cross validation, model building on train set and then verifying performance on test data as mentioned before. Keras library runs by building a ANN model first. We created several such models and compared model output and compute time. We describe the width of such a model by the number of features it uses as neuron for each ANN layer. Similarly, we describe the largeness of such a model as number of hidden layers it contains to compute the output. As our problem is a regression problem, unlike a classical classification problem on ANN, our model has only one node on the output layer (here output, is a number, rather than yes or no values for different classes). Output is the predicted energy simulation result based on the input independent variables 'X' in the model.

ANNs - Experimental Result. Below, we briefly describe the range of models we have experimented to build our model on the training data. For all the models below we used Rectified Linear Unit (RELU) as the activation function. However, we also have tested models with other activation functions like softmax, sigmoid and linear, but RELU's performance was much better than others for our data set.

- Base Model: This model was the simplest of the models studied. It had only one hidden layer. The number of neurons used were 13. This model was much faster than others because of its simplicity and gave accepted result with R2 Score 0.999777.
- Wide Model: This model was bit more complex than the base model. It had only one hidden layer but we increased the number of neurons to 50. This model was slower than the previous with a marginal increment in performance, having output with R2 Score 0.999899.
- Wide and Large Model: This model was the most complex we tried. It tested with 3 hidden layers having over 50 neurons on each layer. This model was way slower than all previous models with a very slight increment in performance, having output with R2 Score 0.999977. Owing to its too much dependence on input features on train data, this model showed signs of overfitting.

The charts in Figures 1 and 2 explain some of the model performance comparison of our ANN Model. After experimenting with different models with changeable parameters, for this dataset, we recommended the baseline model. However, we foresee a scenario, where end users can select the parameters and the model type from Dynamo Node when we query our model engine.



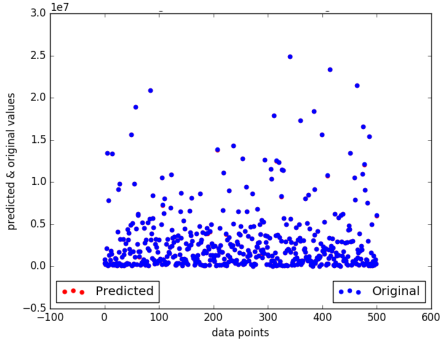


Figure 1
Prediction (x-axis) and simulation (y-axis) results comparison (ANNs)

Figure 2
Prediction and simulation results comparison (y-axis) and data points (x-axis) (ANNs)

Boosted Decision Trees (BDT)

Further, we tested our dataset on Boosted Decision Trees. This method is able to generate predictive models with interpretable tree flowcharts that enable users to quickly understand the model and extract useful information which is an advantage of this method over other widely used techniques (Yu et al. 2010). We used a list of weak learners and iteratively predicted scores on our dataset and tested its accuracy. On each iteration, we upweighted those data points with predicted value was different from actual results. Our final predictor is the weighted average of all the predictors from the weak learners. BDT works by incrementally fixing those data points with predicted score was off from the actual results (Roe et al. 2005).

BDT - Experimental Setup. We used Scikit Learn's DecisionTreeRegressor and AdaBoostRegressor (Pedregosa et al. 2011) to model simple decision trees and boosted decision trees respectively. We followed the same pipeline of cross validation, model building on train set and then verifying performance on test data as mentioned in the previous section.

BDT - Experimental Result.

- Decision tree models: We experimented with simple decision trees with varying maximum depth of pruning for each of our experiments. From model comparison, we realized a maximum depth of 10-12 gave good results. As decision trees were simple yet powerful models, the runtime for our experiment was way faster than ANN models. Our best result from decision tree model was output with a R2 Score of 0.999715.
- Boosted Decision tree models: We experimented with boosted decision trees with varying maximum depth of pruning and varying number of models for the ensemble set for each of our experiments. From model comparison, we realized a maximum depth of 10-12, with 4 models in the ensemble, gave good results. As boosted decision trees were more complex than decision trees, the compute runtime for our experiment was slower than that of decision trees. However, BDT runtime was significantly better than ANN models with almost same performance output. Our best result from BDT model was output with a R2 Score of 0.999985. Figures 3 and 4 show the comparison of the predicted results with energy simulation results for BDT method.

Figure 3
Prediction (x-axis) and simulation (y-axis) results comparison (BDT)

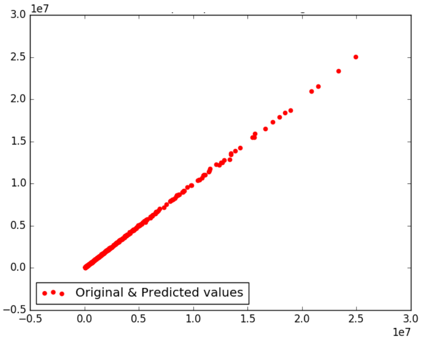
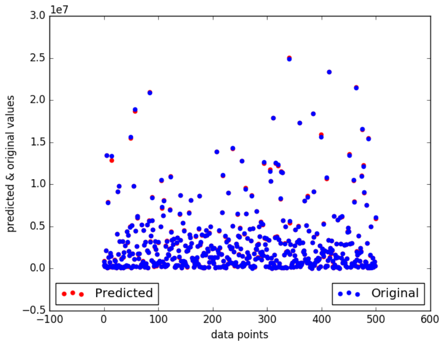


Figure 4
Prediction and simulation results comparison (y-axis) and data points (x-axis) (ANNs)



Feature Selection or Dimensionality Reduction for ANNs and BDT

We tested both ANN and BDT models with feature selection to test if we can decrease model compute runtime without losing performance, especially for ANN. We tested feature selection algorithm Principal Component Analysis (PCA) by setting number of desired features as a model parameter. Out of available 87 features we tested with 20, 30 and 50 needed features. There was little improvement on the runtime, but our results were not as precise as the one we got from the model without feature selection. We also tested other feature selection algorithms like Select K Best and Recursive Feature Elimination (RFE). In some cases, the results were significantly bad and unacceptable with very low R2 score. For this use case and data set, we recommend modelling without using feature selection, but in future use cases, we would like to test feature selection further and refine our model.

CASE STUDY

In order to show the usefulness of the EMM in the early design process, we created a case study of an office building. In this case we use Autodesk Dynamo Studio, a graphical programming interface as the design tool, to parametrically design the model and study its energy performance as a measurement factor. In this study, Window to Wall Ratio (WWR) of North, South, East, West directions as well as Shading Size, Window Width, and Balcony Extension parameters are evaluated as parametric building parameters. All of these parameters are controlling the amount of the lights that enters in the building (balconies of the upper level act as horizontal shading for the lower level). Annual energy use (kBtu) and average WWR are the decision making parameters that are studied in this case. Average WWR is a simple average of the WWRs per direction. To evaluate the energy performance, we pushed the trained model (in this case BDT model) as a service that is accessible by Representational State Transfer (REST) Application Programming Interface (API) as a node in Dynamo Studio. Figure 5 shows the view of Dynamo graph and the geometry in Dynamo interface.

In order to parametrically study the design, Dynamo Studio enables users to publish their model on the web and open the model in project Fractal. Project Fractal enables users to manage exploration of design space providing different generation options and facilitates decision making by Parallel Coordinate Chart (PCC) and visualization of design options in design grid. Figure 6 shows the same model in project Fractal which generated about 7000 design options using cross product generation method.

This case study shows the usefulness of EMM as a energy prediction approximation model which enables design space exploration in a timely manner. Evaluating this large number of options by simu-

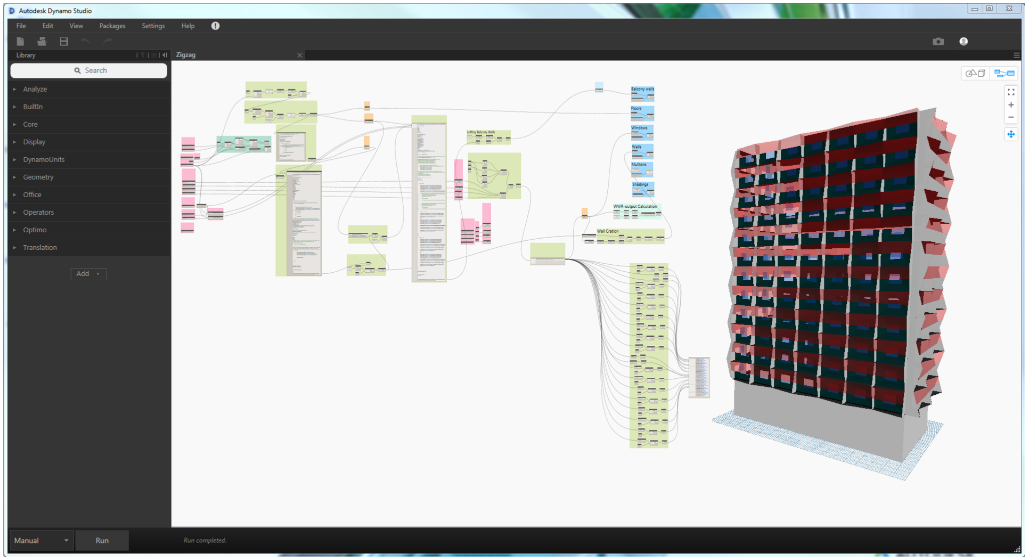


Figure 5
Dynamo graph and the geometry in Dynamo interface

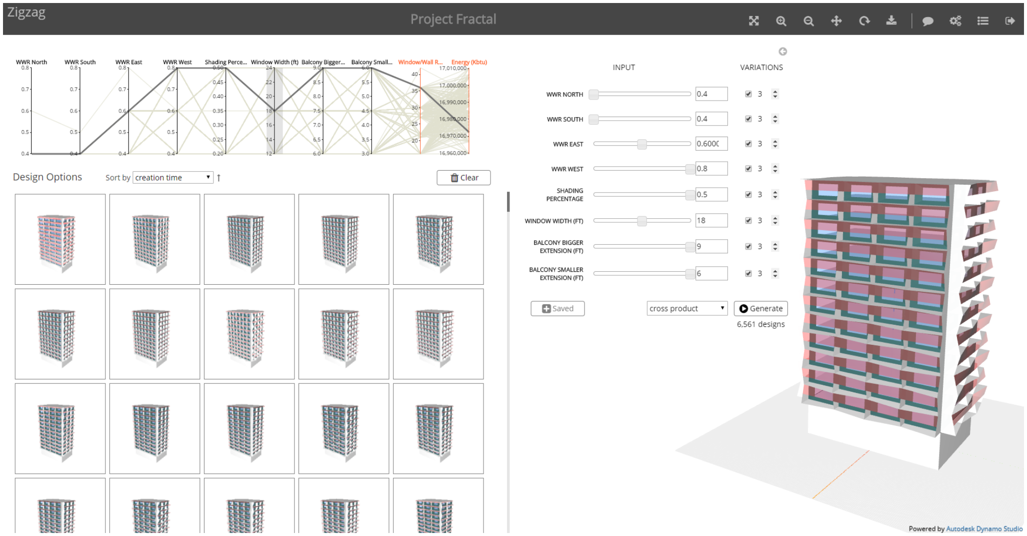


Figure 6
Building model in project Fractal

lating models using conventional energy simulation tools would have taken many days and lots of resources. Using EMM, and other rapid performance analysis factors such as view and comfort factors, the designer can narrow down the design options and then perform detail simulation to get more accurate and detail results.

FUTURE WORK AND CONCLUSION

The objective of the research is to find potential ways to bypass the process of energy simulation for building designers in the early stage of building design. This is mainly for the fact, that energy simulation takes a significant amount of time, is resource intensive and needs a certain level of detail in the building model, which is usually low in early stages. EMM uses an extensive set of parameters to track all of the impactful variables on energy performance of building model and learns from existing data by using ML algorithms providing quick and real-time energy simulation feedback to designers. These parameters cover geometry, construction, and load related building variables. Considering these parameters from all of the three categories mentioned above in the training process of the machine learning methods and making the service available to be easily used in design applications is the main contribution of this paper.

Current implementation of EMM uses ANNs since it can predict accurate results for complex interdependent problems. EMM also includes BDT since it enables users to quickly understand the model and extract useful information which is very useful for architects. As it was demonstrated in the body of the manuscript, both models have acceptable accuracy in predicting the energy performance factor. EMM makes the trained model available as a service and make it easy to be accessed by any parametric design tool. The user would only need to call the service through the provided API and add it to the design process.

The case study of the office building model design exploration using two of the common parametric design tools demonstrates the usefulness of this approach. Using EMM we were able to explore about 7000 building design options and their energy performance and make informed design decision in the conceptual design phase.

As part of the future work, we are also adding multiple images of the building model from various angles as training parameters to be able to track the geometry of the building model more accurately. We are training the models on the top of these images to categorize building model geometry and increase the accuracy of the results. One of the drawbacks of the current system implementation is that it is feeding all of the features in the dataset into the ML Model for prediction. However, from expert's domain knowledge, we know every parameter of the building design does not have the same impact in building energy use computation. Thus, our in-progress work includes implementing feature selection and dimensionality reduction algorithms to the data set, before we pass it on to the selected ML algorithm.

ACKNOWLEDGEMENT

We would like to acknowledge and thank Omid Oliyan Torghabehi for his work and support on preparing the case study for this paper.

REFERENCES

Ansari, F, Mokhtar, A, Abbas, K and Adam, N 2005, 'A simple approach for building cooling load estimation', *Am. J. Environ. Sci*, 1(3), pp. 209-212

Asl, MR, Zarrinmehr, S, Bergin, M and Yan, W 2015, 'BPOpt: A framework for BIM-based performance optimization', *Energy and Buildings*, 108, pp. 401-412

Bauer, M and Scartezzini, JL 1998, 'A simplified correlation method accounting for heating and cooling loads in energy-efficient buildings', *Energy and Buildings*, 27(2), pp. 147-154

Catalina, T, Virgone, J and Blanco, E 2008, 'Development and validation of regression models to predict monthly heating demand for residential buildings', *Energy and buildings*, 40(10), pp. 1825-1832

DOE, US 2012, *Building energy software tools directory*, Department of Energy

Dong, B, Cao, C and Lee, SE 2005, 'Applying support vector machines to predict building energy consump-

tion in tropical region', *Energy and Buildings*, 37(5), pp. 545-553
Dounis, AI 2010, 'Artificial intelligence for energy conservation in buildings', *Advances in Building Energy Research*, 4(1), pp. 267-299
Guo, X, Li, W and Iorio, F 2016 'Convolutional neural networks for steady flow approximation', *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 481-490
Hopfield, JJ 1982, 'Neural networks and physical systems with emergent collective computational abilities', *Proceedings of the national academy of sciences*, 79(8), pp. 2554-2558
Hunter, JD 2007, 'Matplotlib: A 2D graphics environment', *Computing In Science \& Engineering*, 9(3), pp. 90-95
Krarti, M 2003, 'An overview of artificial intelligence-based methods for building energy systems', *TRANSACTIONS-AMERICAN SOCIETY OF MECHANICAL ENGINEERS JOURNAL OF SOLAR ENERGY ENGINEERING*, 125(3), pp. 331-342
Li, Q, Meng, Q, Cai, J, Yoshino, H and Mochida, A 2009, 'Applying support vector machine to predict hourly cooling load in the building', *Applied Energy*, 86(10), pp. 2249-2256
Pedregosa, F, Varoquaux, G, Gramfort, A, Michel, V, Thirion, B, Grisel, O, Blondel, M, Prettenhofer, P, Weiss, R and Dubourg, V 2011, 'Scikit-learn: Machine learning in Python', *Journal of Machine Learning Research*, 12(Oct), pp. 2825-2830
Pour, ZA and Ayat, S 2014, 'Comparison between artificial neural network learning algorithms for prediction of student average considering effective factors in Learning and educational progress', *Journal of Mathematics and Computer Science*, 8, pp. 215-225
Roe, BP, Yang, HJ, Zhu, J, Liu, Y, Stancu, I and McGregor, G 2005, 'Boosted decision trees as an alternative to artificial neural networks for particle identification', *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 543(2), pp. 577-584
Tsanas, A and Xifara, A 2012, 'Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools', *Energy and Buildings*, 49, pp. 560-567
Walt, Svd, Colbert, SC and Varoquaux, G 2011, 'The NumPy array: a structure for efficient numerical computation', *Computing in Science \& Engineering*, 13(2), pp. 22-30
Yu, Z, Haghighat, F, Fung, BC and Yoshino, H 2010, 'A decision tree method for building energy demand modeling', *Energy and Buildings*, 42(10), pp. 1637-1646
Zhang, J and Haghighat, F 2010, 'Development of Artificial Neural Network based heat convection algorithm for thermal simulation of large rectangular cross-sectional area Earth-to-Air Heat Exchangers', *Energy and Buildings*, 42(4), pp. 435-440
Zhao, Hx and Magoulès, F 2012, 'A review on the prediction of building energy consumption', *Renewable and Sustainable Energy Reviews*, 16(6), pp. 3586-3592