2022 European Conference on Computing in Construction
Ixia, Rhodes, Greece
July 24-26, 2022

ENERGY-AWARE DESIGN: PREDICTING BUILDING PERFORMANCE FROM LAYOUT GRAPHS

Jianpeng Cao[1], Hang Zhang[2], Anton Savov[1,2], Daniel Hall[1] and Benjamin Dillenburger[2]
[1]ETH Zurich, Institute of Construction and Infrastructure Management, Zurich, Switzerland
[2]ETH Zurich, Institute of Technology in Architecture, Zurich, Switzerland

Abstract

Graph Neural Networks (GNNs) have become a popular toolkit for generative floor plan design. Although design variation has improved greatly, few studies consider non-geometrical characteristics, such as building performance, in the generative design process. This paper presents a GNN-based approach to predict the energy performance for floor plan customization (energy-aware design). The approach lays the foundation for a performance-aware generative design using GNN. The results show that the GNN can achieve high accuracy in energy performance prediction.

Keywords: GNN, spatial layout, building energy performance

1. Introduction

The European 2030 climate and energy framework sets targets for cutting greenhouse gas emissions and increasing the share of renewable energy and energy efficiency (European Commission, 2020). Building accounts for 30%-40% of total energy use globally (Programme United Nations Environment, 2007), and hence, building sectors have a large potential to reduce energy consumption. It is crucial to conduct energy analysis at the design stage to ensure future buildings are more energy-efficient. Decisions made at the early design stage can save up to 30% energy usage with no added cost (Elbeltagi et al., 2017).

Graph-based generative design techniques can achieve high customization and efficiency in the schematic design of buildings (Hu et al., 2020; Para et al., 2020). However, existing generative design approaches on floorplan design seldom consider building performance, such as energy efficiency. Only after clients select one or a few designs the building performance is evaluated using a traditional simulation tool. For example, the energy performance can be simulated using EnergyPlus (Department of Energy, 2021). This process is too slow for real-time feedback. Furthermore, the tools require the user to specify large sets of parameters which are usually uncertain at the early design stage. Overall, relying on simulation for performance estimating is time-intensive (Pham et al., 2020). Therefore, a new approach is needed to integrate building performance estimates into generative design techniques.

Data-driven approaches for building performance prediction use machine learning algorithms. Those algorithms include Decision Tree (DT) (Yu et al., 2010), Neural Network (NN) (Biswas, Robinson and Fumo, 2016), Random Forest (RF) (Wang et al., 2018), Support Vector Machine (SVM) (Li et al., 2009) and multiple regression model (Catalina, Iordache and Caracaleanu, 2013). Compared with the simulation models, those data-driven approaches evaluate design with less information and generate results in a shorter time (Qiao, Yunusa-Kaltungo, and Edwards, 2021).

In pursuit of energy-aware generative design, we develop a GNN-based approach to building performance prediction. Although many studies have reviewed data-driven predictive models for building energy performance (Foucquier et al., 2013; Zhang et al., 2021), no study has investigated GNN-based energy prediction. Furthermore, previous approaches do not take spatial layouts as a design variable (Li et al., 2009; Yu et al., 2010; Wang et al., 2018; Olu-Ajayi et al., 2022), since they assume that the design is fixed when conducting energy analysis. However, the spatial layout is an important design variable for building energy performance (Du et al., 2021), and architects adjust it iteratively during the early design phase.

To address this question, this paper is structured as follows. In Section 2, an overview of data-driven energy prediction and GNN-based floorplan design are given. In Section 3, a three-step approach for building performance prediction with GNN is presented. Three criteria are formulated to choose among existing GNN architectures based on the nature of spatial layout graphs. In Section 4, a test case with the proposed approach is implemented. In Section 5, a sensitivity analysis to check the dependence between floor layouts and building energy performance is conducted. Finally, the approach's limitations and directions for future work are discussed.

2. Literature review

In this section, we explore the potential to predict energy performance for floorplan design using GNN. We review two related topics: data-driven energy prediction models and GNN-based floor plan design. In the review of the first topic, we focus on the input features and data structures. For the GNN review, we pay attention to the graph representation of the design.

2.1. Data-driven energy prediction model

The energy performance of buildings can be influenced by many factors, which can be categorized into four categories: (i) weather data, (ii) building data, (iii) the operation data, and (iv) occupancy data (Zhao and Magoulès, 2012). The second category is the most relevant for us as layout graphs represent building data.

For example, Wang et al. predicted building-level electricity usage using Random Forest (Wang et al., 2018). The model takes ambient weather conditions, occupancy data, and operation time as inputs, but building structure and characteristics are neglected. Yu et al. developed a decision tree model that considers all four categories above in predicting annual energy use intensity (Yu et al., 2010). The building characteristics include house type, construction type, floor area, heat loss coefficient, and leakage area. However, heat loss coefficient and leakage area are not accessible at the early design stage. Olu-Ajayi et al. compared nine machine learning algorithms to predict a building's annual energy rating (Olu-Ajayi et al., 2022). The input features, including weather and building data, are collected at the early design phase. The building data consists of floor area, glazed area, house type, building element description (walls, windows, and roofs). The review by Zhang et al., 2021) presents further building-related data used for energy prediction models.

However, none of the studies using the second category, building data, use the spatial layout to predict performance. One possible reason can be that previous work represents input features as n-dimensional vectors, which cannot fully represent spatial layouts. Unrelated to data representations, one more possible reason is that spatial layout might not be a strong predictor of building performance.

In our work, we use GNNs that take graphs as input and can open up the possibility of exploring if spatial layout data can be used in performance prediction.

2.2. GNN-based floorplan design

In the field of architecture, the spatial relations between each room or structural elements in the layout are traditionally represented by 2D floor plan drawings. Similarly, many generative design techniques have represented layouts as pixel images (Huang and Zheng, 2018) or voxelized wireframes (Miguel et al., 2020). However, these data types lose adjacency values between each layout element. Benefiting from the recent development in artificial intelligence, especially the graph-based deep learning algorithms, several previous works use graph neural networks (GNNs) for floor plan generation. Nauata et al., 2020 generate house layouts using constrained graphs. Para et al. took a further step than using constraint graphs by considering additional features, such as the number of bedrooms, for output manipulation (Para et al., 2020). Based on the given adjacency graph, Shekhawat et al. realized the floorplan generation with the room dimension defined by the user (Shekhawat et al., 2021). Zhang, 2020 used GNN to generate architectural layouts from a user's text description, parsing it into graphs with a linguistic parser. Hu et al., 2020 make the adjusted-graph input for their GNN floor plan generation open to the user for instant editing.

However, all of these related works are confined to simple architectural layout design. No objective functions have been considered, such as structural performance, construction cost, energy behavior, etc. The prediction of energy performance based on plan layout has been implemented through several different types of data structures. For instance, Wortmann and Natanian realized the multi-objective (including energy demands for heating and cooling) optimization for zero-energy urban design with data of building block shapes (Wortmann and Natanian, 2020). And also, a deep reinforcement learning model was proposed to optimize the solar energy performance of the building's layout in a defined district (Han, Yan, and Liu, 2020). Nevertheless, this previous research simply laid on the general plan shapes and did not involve the inner rooms layout or the spatial relations of plan elements in the analysis. As a result, there is an opportunity to use GNN to develop a performance-aware generative design for the architectural floorplan.

3. Proposed approach

In this section, we propose a GNN-based approach for the building's energy performance prediction. The approach is composed of the following steps:

1. Dataset generation
2. Graph representation of an architectural design
3. Building performance prediction by GNN

3.1. Dataset generation

We use a generative design tool and an energy simulation engine to generate the input and output of the training data for building performance prediction by GNN. The generative design tool enables designers to specify design parameters and generate all possible solution permutations. The parameters are set within a feasible range, and the correlation of parameters can be defined as constraints. For example, a floorplan generator can create diverse floor layouts by initializing parameters, such as space type and locations automatically. Once the design alternatives are completed, the selected energy simulation engine calculates the performance values. Given a site condition, the weather data is defined in an EPW (EnergyPlus Weather Format) and set as a constant for all alternatives. The simulation results include heating load, cooling load, total energy consumption, etc., which can be encoded into a performance matrix.

3.2. Graph representation of an architectural design

The input of the GNN model is a graph representation of an architectural design. At the early design stage, the design can be a layout composed of various functional spaces or main building elements. Take a space as an example; the space relates to another space in different patterns which define their interaction. Typical space relationships include adjacent spaces, interlocking spaces, space within a space, spaces linked by a common space (Francis D. K. Ching, 2007; Hillier, 2007). Each space is defined by a set of features, such as room type, area, location, etc. A feature matrix and an adjacency matrix represent such a design as a graph. The feature matrix is a *m x n* matrix, where *n* equals the number of spaces and *m* equals the number of features used to describe a space. The feature matric represents the features for each space. The adjacency matrix represents relationships between spaces as a *n x n* matrix. In the data structure of the adjacency matrix, value 1 indicates the rooms to be connected, while value 0 refers to no connection. Figure 1 shows an example of floorplan design and corresponding feature and adjacency matrices. Figure 2 shows the energy performance of the given floorplan.

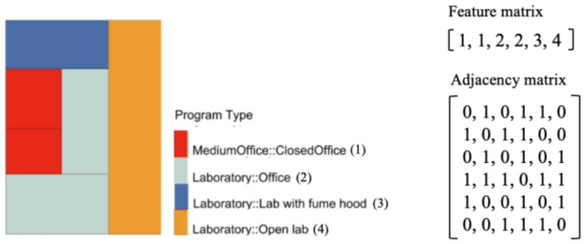

Figure 1: A floorplan design (left) and 1 × 6 feature matrix (right up) and 6 × 6 adjacency matrix (right down)

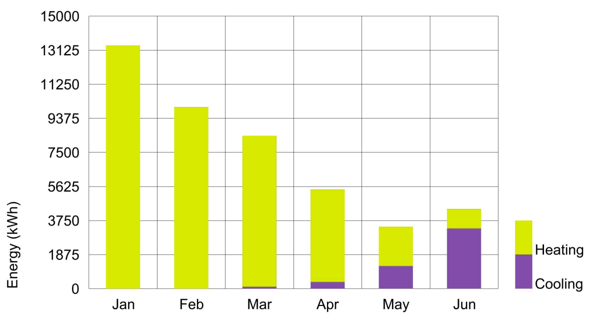

Figure 2: Energy performance for the given floorplan

3.3. Building performance prediction by GNN

Graph neural networks are standardly designed to process and analyze data represented as graphs. For this research, the energy performance prediction is regarded as a graph regression problem (Dwivedi et al., 2020). In other words, the task is to regress the output variable, namely energy consumption, against the given graph structure. To guide our choice among the existing GNN models for graph regression (Dwivedi et al., 2020), including GCN (Kipf and Welling, 2017), GraphSage (Ahmed et al., 2017), MoNet (Monti et al., 2017), GAT (Veličković et al., 2018), and GatedGCN (Bresson and Laurent, 2017), we consider the following three criteria:

1. Is the type of the space relationship (edge feature) included or not?

For instance, the types of spatial relationships, such as interlocking spaces, can be modeled as edge features of the input graph. GCN and GraphSage, for example, do not consider incorporating edge features in their update equation. Therefore, they cannot distinguish between designs with the same number and size rooms but different spatial relationships between them.

2. Is the degree of spatial connectivity high or low (sparse or dense graph)?

In mathematics, a dense graph has an edge count close to the maximal number of edges. In architectural design, it refers to a design in which each space is connected to many adjacent spaces. GNN architectures consisting of a long short-term memory (LSTM) based neighborhood aggregation mechanism, such as GatedGCN, perform poorly in dense graphs. Therefore, it is not efficient for a design with a high degree of spatial connectivity.

3. Is the scale of design, e.g., the graph size, large or small?

The information propagation and aggregation in the graph require a long computation time when the graphs are large. GNN models with a sampling module, like GraphSage, can mitigate the problem. Another solution is to include an LSTM layer in the model, such as GatedGCN. However, this criterion becomes less important with the increase in computing power.

Table 2: Suitability of GNNs based on the nature of floor plan layout graphs

	consider types of spatial relationships	high degree of spatial connectivity	large number of spaces
GCN	Not support	Suitable	Not suitable
GraphSage	Not support	Suitable	Suitable
MoNet	Support	Suitable	Not suitable
GAT	Support	Suitable	Not suitable
GatedGCN	Support	Not suitable	Suitable

Based on the comparison above, given the fact that our graph-based representation of design is a layout graph with no edge features, all GNNs are suitable for testing the approach.

4. Implementation

The general workflow is illustrated in Figure 3. Building energy analysis is mainly processed through s1.3.0, which are based on the Grasshopper environment in Rhino 7. Construction sets are defined with cool climate zone, building vintage of IECC 2015, and steel-framed construction type. The generated designs consist of six rooms with fixed sizes (10,15,20,25,30,35 in square meters) and height (3 meters). At the same time, their corresponding positions and shapes on the floor plan are randomized via the Grasshopper add-on Marmot. This addon transfers the graph-based information into a schematic floor plan. In our study, each room could have six possible types: office, writeup, lab support, laboratory, conference, and classroom. Their energy performance parameter is pre-defined using the Honeybee energy library. The ratio between the area of the apertures and the area of the parent face is set to 35%, the height of the

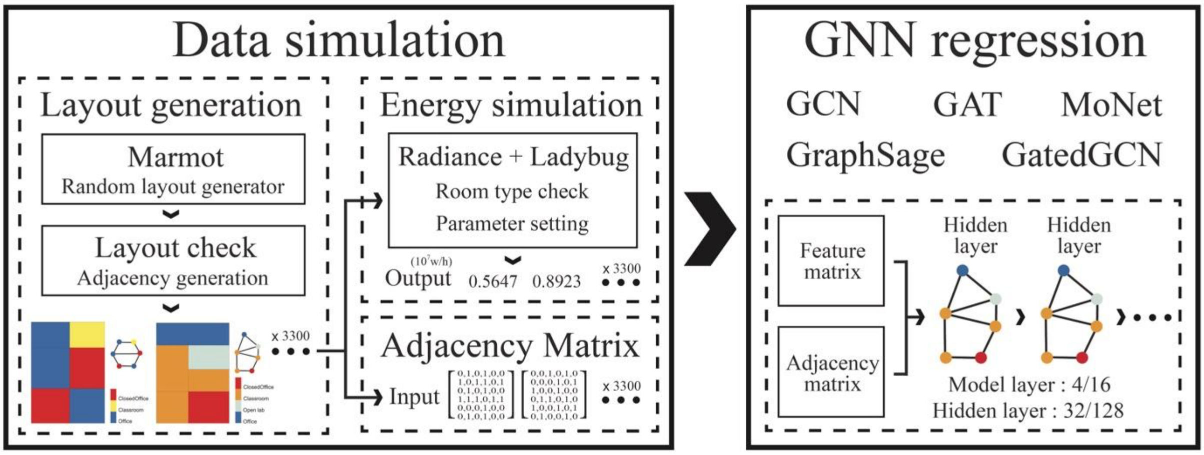

Figure 3: The workflow of the proposed approach

window is 2.8 meters, the effectiveness of sensible heat recovery is 0.81, and latent heat recovery is 0.75 (Tang, Ahmad and Yusup, 2018).

Our energy simulation will take for a whole annual period (with the sampling every 4 hours) at the region of logan in Boston (in energy plus weather format). Based on our computing configuration (AMD Ryzen 3900X), we generated and simulated a total of 3300 cases (taking 21 days) with our energy analysis process and recorded the results into three separate matrices for the graph neural network training. The feature matrix stores the room type information, the adjacency matrix represents the spatial relations of the floorplan layout, and the performance matrix offers the final result of energy analysis. To narrow the output dimension for easier training in the neural network, we merge the energy consumption of 12 months into one mean value.

The graph neural network infrastructure is built on PyTorch (Paszke et al., 2019) with the Adam optimizer (Kingma and Ba, 2015) and MAE (mean absolute error) as loss function. The data set is shuffled and split into training, testing, and validating set with the ratio of 8:1:1. The function of the scheduled learning rate is implemented so that the rate keeps lowering once there is no improvement of the validation loss in 5 epochs. A maximum of 1000 epochs with the initial learning rate of 0.001 is configured, and the training will break out once the current learning rate is lower than the baseline we defined, which is 0.000001.

In this study, several graph-based machine learning algorithms are implemented to test our data structure, such as GCN, MoNet, GAT, GraphSage, and GatedGCN. In all of these implementations, the residual connections (He et al., 2016) are embedded in the forward function, and each of the message-passing GCN layers will add batch normalization. In the final layers of these networks, a three layers downstream MLP (Multilayer perceptron) is built to process the final output. We set the kernel size of the MoNet model to 3, and the number of heads in the GAT model to 8.

Table 1 : Results of regression with variant GNN configuration

Model	Hidden layer	Model layer	Parameters	Epochs	Time	Test MAE	Train MAE
GatedGCN	32	4	22785	136	4.1	0.1618	0.1682
GatedGCN	32	16	87681	128	13.2	0.1618	0.1687
GatedGCN	128	4	344577	123	5.5	0.1618	0.1692
GatedGCN	128	16	1341441	147	22.9	0.1618	0.1671
GCN	32	4	5377	355	4.0	0.1621	0.1675
GCN	32	16	18817	308	9.2	0.1619	0.1677
GCN	128	4	78337	298	4.4	0.1619	0.166
GCN	128	16	279553	313	13	0.1621	0.1671
MoNet	32	4	13641	389	7.2	0.1626	0.1675
MoNet	32	16	51873	445	24.9	0.1626	0.1667
MoNet	128	4	209481	244	7.7	0.1623	0.1662
MoNet	128	16	804129	339	35.5	0.1624	0.167
GAT	32	4	309249	368	17.0	0.1621	0.1673
GAT	32	16	1107969	348	58.4	0.1618	0.1669
GAT	128	4	4874241	314	51.9	0.162	0.1678

Model	Hidden layer	Model layer	Parameters	Epochs	Time	Test MAE	Train MAE
GatedGCN	32	4	22785	136	4.1	0.1618	0.1682
GAT	128	16	17506305	380	235.5	0.1618	0.1679
GraphSage	32	4	9473	295	4.9	0.1621	0.1682
GraphSage	32	16	35201	310	5.1	0.1619	0.1671
GraphSage	128	4	143873	345	6.3	0.1618	0.1674
GraphSage	128	16	541697	394	19.5	0.1618	0.1666

All of these networks use library DGL (Wang et al., 2019) to process the graph data, with the environment of CPU on Windows PyTorch, and be trained via AMD Ryzen 3900X. As table 1 shows, variant configurations of hidden layer numbers and batch size scale has been tested. We've evaluated the performance of these networks with MAE (mean absolute error) in both the training set and testing set. As the MAE calculates the distance between the ground truth and predicted value, the goal is to lower the MAE to near 0. As a result, the table indicates that all of the tested graph-based models have a stable and decent performance of our defined energy prediction task.

5. Discussions

The prediction model is built upon the presumption that the floor layout design affects the building energy performance. Although previous scholars have proven the statement (Du et al., 2021), this study also conducts a sensitivity analysis to validate the presumption based on our simulation platform (Ladybug + Radiance). In order to eliminate the effect of other variants (room size, room shape, and room type), each room is fixed as a 3m*3m dimension with the same thickness of the walls. The study was also conducted with the same number of room types, whose simulation attributes are pre-defined in Radiance, and color tagged as: red for office, light yellow for writeup, dark yellow for conference, light orange for laboratory, dark orange for lab support, light blue for classroom, medium blue for corridor, and dark blue for storage. A total of 100 test cases were simulated, while the only differences between them is the spatial layout that is randomly generated.

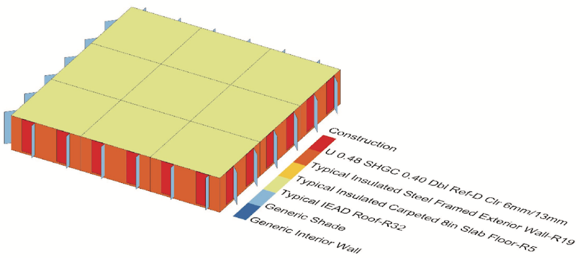

Figure 4: Energy simulation model

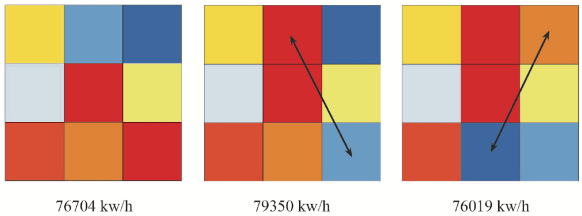

Figure 5: Layout swap and corresponding building energy performance output

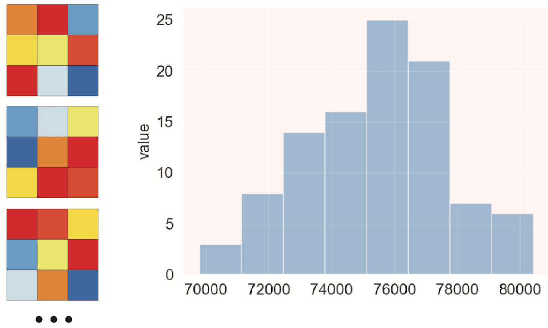

Figure 6: Energy distribution of 100 test layouts

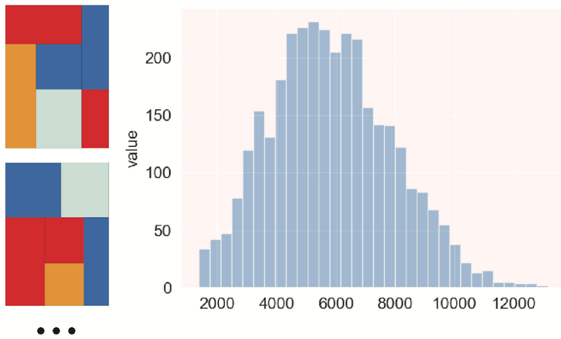

Figure 7: Energy distribution of 3300 simulation layouts

Figure 4 illustrates the simulation model with the default setting in Grasshopper (including window height, sizes, etc.). With the definition of orientation (North, South, etc.), the solar condition and louver settings (rotation degree, depth, etc.) on each side of the facade will have corresponding different parameters. It will help to explain the different energy outputs in Figure 5. Figure 5 shows the results when swapping the location of 2 rooms. The energy analysis results will change with the modified room layout (same as the editing of adjacency graph), while the general room types and numbers are fixed. Figure 6 is the distribution of energy behavior with 100 test experiments. The room types and numbers are fixed (meaning the 'Feature matrix' is set to constant), as a result, it concentratedly indicates the adjacency influence on the simulation results. Figure 7 shows the diversity of the 3300 simulation data in this paper. With more freedom on the 'Feature matrix', the distribution of the results gets higher diversity.

6. Limitations

However, the prediction model has two main limitations. First, the variables influencing energy performance are not fully taken into the prediction model. This work focuses on the design variables, such as spatial layouts and space types, defined by architects and clients at the early phase. Many other variables, such as weather data, are not included in the model. Another reason that influences the accuracy of the prediction performance is the GNN model itself. The message-passing-based GNNs are used, while Weisfeiler-Lehman based GNNs (WLGNN) that are not scalable for large datasets (Dwivedi et al., 2020) are not included in this work. New GNN models might be applied to take more design variables into account.

Second, there is a lack of an interface that can integrate generative floorplan design and energy performance modeling. The study takes the generated floorplans as input and proceeds it via a GNN prediction model to get the performance output. Considering that the floorplans can be modeled in different approaches, such as 3D models or 2D sketches, manual data processing is done in this work to convert the design into the graph format. This can be solved by applying an object detection algorithm to automatically identify the design elements.

7. Conclusions

This paper makes three main contributions to data-driven energy prediction. Firstly, the proposed approach considers spatial layout a novel variable in the energy prediction model. Based on the test case presented in the implementation section, we conclude the approach shows good performance and potential to be scalable to more architectural variables, such as room locations. Second, a GNN model is utilized for the first time to predict a building's energy performance. Using graphs as prediction inputs lays the foundation to make the newly emerging GNN generative design techniques performance-aware. And third, the proposed approach defines three criteria in model selection according to design characteristics. By doing so, we not only guide domain experts in AI-based applications but also explore a new research direction in matching machine learning algorithms with design characteristics.

Future research will pursue three goals. First, more architectural features, such as room area, room locations, space relationship types, to the existing GNN model can be added. The structure of the GNN model needs to be adapted to multi-dimensional features accordingly. Second, the performance of the GNN models can be analyzed according to graphs with different characteristics: size, density, etc. Last but not least, we aim to investigate whether GNNs can be used to find the design elements that influence the given building performance, e.g., energy efficiency, the most.

References

Ahmed, N. K., Rossi, R. A., Zhou, R., Lee, J. B., Kong, X., Willke, T. L., & Eldardiry, H. (2017) Inductive Representation Learning in Large Attributed Graphs, arXiv preprint arXiv:1710.09471.

Biswas, M. A. R., Robinson, M. D. and Fumo, N. (2016) Prediction of residential building energy consumption: A neural network approach, *Energy*, 117, pp. 84–92.

Bodnar, C., Frasca, F., Otter, N., Wang, Y. G., Liò, P., Montufar, G. F., & Bronstein, M. (2021) Weisfeiler and Lehman Go Cellular: CW Networks, *Advances in Neural Information Processing Systems*, 34, pp. 1–28.

Bresson, X. and Laurent, T. (2017) Residual Gated Graph ConvNets, arXiv preprint arXiv:1711.07553.

Catalina, T., Iordache, V. and Caracaleanu, B. (2013) Multiple regression model for fast prediction of the heating energy demand, *Energy and Buildings*, 57, pp. 302–312.

Department of Energy (2021) EnergyPlus. viewed Jan 2022, <https://energyplus.net/>.

Du, T., Jansen, S., Turrin, M., & van den Dobbelsteen, A. (2021) Effect of space layouts on the energy performance of office buildings in three climates, *Journal of Building Engineering*, 39(January), p. 102198. doi: 10.1016/j.jobe.2021.102198.

Dwivedi, V. P., Joshi, C. K., Laurent, T., Bengio, Y., & Bresson, X. (2020) Benchmarking Graph Neural Networks, arXiv preprint arXiv:2003.00982.

Elbeltagi, E., Wefki, H., Abdrabou, S., Dawood, M., & Ramzy, A. (2017) Visualized strategy for predicting buildings energy consumption during early design stage using parametric analysis, *Journal of Building Engineering*, 13(July), pp. 127–136.

European Commission (2020) 2030 climate & energy framework, viewed Jan 2022, <https://ec.europa.eu/clima/eu-action/climate-strategies-targets/2030-climate-energy-framework_en>.

Foucquier, A., Robert, S., Suard, F., Stéphan, L., & Jay, A. (2013) State of the art in building modelling and energy performances prediction: A review, *Renewable and Sustainable Energy Reviews*, 23, pp. 272–288.

Francis D. K. Ching (2007) *Architecture: Form, Space, and Order*. John Wiley & Sons.

Han, Z., Yan, W. and Liu, G. (2020) A Performance-Based Urban Block Generative Design Using Deep Reinforcement Learning and Computer Vision, *in The International Conference on Computational Design and Robotic Fabrication*, pp. 134–143.

He, K., Zhang, X., Ren, S., & Sun, J. (2016) Deep residual learning for image recognition, *Proceedings of the*

IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-December, pp. 770–778.

Hillier, B. (2007) Space is the machine, Design Studies.

Hu, R., Huang, Z., Tang, Y., Van Kaick, O., Zhang, H., & Huang, H. (2020) Graph2Plan: Learning Floorplan Generation from Layout Graphs, *ACM Transactions on Graphics*, 39(4), pp. 118-1.

Huang, W. and Zheng, H. (2018) Architectural drawings recognition and generation through machine learning, *Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture*, pp. 156–165.

Kingma, D. P. and Ba, J. L. (2015) Adam: A method for stochastic optimization, 3rd International Conference on Learning Representations, *ICLR 2015 - Conference Track Proceedings*, pp. 1–15.

Kipf, T. N. and Welling, M. (2017) Semi-supervised classification with graph convolutional networks, *5th International Conference on Learning Representations*, pp. 1–14.

Li, Q., Meng, Q., Cai, J., Yoshino, H., & Mochida, A. (2009) Applying support vector machine to predict hourly cooling load in the building, *Applied Energy*, 86(10), pp. 2249–2256.

Miguel, J. de, Villafane, M.E.; Piskorec, L.; Sancho-Caparrini, F. (2020) Deep Form Finding Using Variational Autoencoders for deep form finding of structural typologies, *Architecture in the Age of the 4th Industrial Revolution - Proceedings of the 37th eCAADe and 23rd SIGraDi Conference*, pp. 71–80.

Monti, F., Boscaini, D., Masci, J., Rodola, E., Svoboda, J., & Bronstein, M. M. (2017) Geometric deep learning on graphs and manifolds using mixture model CNNs, Proceedings - *30th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5425–5434.

Nauata, N., Chang, K. H., Cheng, C. Y., Mori, G., & Furukawa, Y. (2020) House-GAN: Relational Generative Adversarial Networks for Graph-constrained House Layout Generation, *in European Conference on Computer Vision*, pp. 162–177.

Olu-Ajayi, R., Alaka, H., Sulaimon, I., Sunmola, F., & Ajayi, S. (2022) Machine learning for energy performance prediction at the design stage of buildings, *Energy for Sustainable Development*, 66, pp. 12–25.

Para, W., Guerrero, P., Kelly, T., Guibas, L. J., & Wonka, P. (2020) Generative Layout Modeling using Constraint Graphs, *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6690–6700.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L. and Desmaison, A. (2019) PyTorch: An imperative style, high-performance deep learning library, *Advances in Neural Information Processing Systems*, 32, pp.8026-8037.

Pham, A. D., Ngo, N. T., Truong, T. T. H., Huynh, N. T., & Truong, N. S. (2020) Predicting energy consumption in multiple buildings using machine learning for improving energy efficiency and sustainability, *Journal of Cleaner Production*, 260, p. 121082.

Programme United Nations Environment (2007) Buildings and Climate Change: Current Status, Challenges and Opportunities, viewed Jan 2022, <https://wedocs.unep.org/handle/20.500.11822/7783>.

Qiao, Q., Yunusa-Kaltungo, A. and Edwards, R. E. (2021) Towards developing a systematic knowledge trend for building energy consumption prediction, *Journal of Building Engineering*, 35, p. 101967.

Shekhawat, K., Upasani, N., Bisht, S., & Jain, R. N. (2021) A tool for computer-generated dimensioned floorplans based on given adjacencies, *Automation in Construction*, 127(August 2020), p. 103718.

Tang, C.Y., Ahmad, M.I. and Yusup, Y., 2018. Development and Performance Investigation of Energy Recovery System in Tropical Climate. *Journal of Physical Science*, 29(1), pp.17-30.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2018) Graph attention networks, *6th International Conference on Learning Representations*, pp. 1–12.

Wang, M., Yu, L., Zheng, D., Gan, Q., Gai, Y., Ye, Z., Li, M., Zhou, J., Huang, Q., Ma, C. and Huang, Z. (2019) Deep Graph Library: Towards Efficient and Scalable Deep Learning on Graphs, *Proceedings of the ICLR Workshop on Representation Learning on Graphs and Manifolds*, pp. 1–7.

Wang, Z., Wang, Y., Zeng, R., Srinivasan, R. S., & Ahrentzen, S. (2018) Random Forest based hourly building energy prediction, *Energy and Buildings*, 171, pp. 11–25.

Wortmann, T. and Natanian, J. (2020) Multi-Objective Optimization for Zero-Energy Urban Design in China: A Benchmark, *Proceedings of the Symposium on Simulation for Architecture and Urban Design*, (May), pp. 203–210.

Yu, Z., Haghighat, F., Fung, B. C., & Yoshino, H. (2010) A decision tree method for building energy demand modeling, *Energy and Buildings*, 42(10), pp. 1637–1646.

Zhang, H. (2020) Text-to-Form, *Proceedings of the 40th Annual Conference of the Association for Computer Aided Design in Architecture*, pp. 238–247

Zhang, L., Wen, J., Li, Y., Chen, J., Ye, Y., Fu, Y., & Livingood, W. (2021) A review of machine learning in building load prediction, *Applied Energy*, 285(January), p. 116452.

Zhao, H. X. and Magoulès, F. (2012) A review on the prediction of building energy consumption, *Renewable and Sustainable Energy Reviews*, 16(6), pp. 3586–3592.