Problem Statement

Architecture is a complex subject that requires a vast number of knowledge and experience. Both stability and beauty need consideration. Machine Learning (ML) is a technique that helps machines discover the best solution by training from countless data. Theoretically, if the machine gets trained with enough data and rounds, it can form a perfect prediction model. However, the fact is that people only have limited data, and they prefer to have a trade-off between the model precision and the training cost. Therefore, how to apply ML efficiently in building architecture, such as designing a floor plan and decorating rooms, is what we want to realize as engineers to power the world.

Application

The first kind of application is related to architectural stability. Architects need to consider several environmental factors when they select building materials. Such as temperature, humidity, sunshine duration, and so on. Materials perform differently in a single condition. However, in the real world, in a complicated condition, it is hard to say which one has the best performance. Architects can only make decisions from their past experiences. It is the same principle as Machine Learning. With a well-trained model, architects only need to input local environmental factors and wait for the best choice in that condition.

Another kind of application is derived from architectural aesthetics. Although different people indeed have different aesthetics, there still exist several similarities. Those are so-called public aesthetics. For architecture companies, designing a satisfying plan for customers is the most difficult and important first step. Nevertheless, if they have owned enough data and trained an available model by machine learning, they can then promptly use the machine to produce several basic plans according to customers' requirements. Customers can choose from them or ask for more. After deciding on the basic plan, companies then make customized changes to satisfy any specific preference. With the help of machine learning, it becomes easier for decorating companies to understand their customers' requirements and make deals.

Related Work

Building energy consumption prediction for residential buildings using deep learning and other machine learning techniques, https://www.sciencedirect.com/science/article/abs/pii/S235271022101264X

Component-based machine learning for performance prediction in building design, https://www.sciencedirect.com/science/article/pii/S0306261918310389

Construction safety management in the data-rich era: A hybrid review based upon three perspectives of nature of dataset, machine learning approach, and research topic, https://www.sciencedirect.com/science/article/abs/pii/S1474034623002720

Imbalanced Data Classification Based on Extreme Learning Machine Autoencoder, https://ieeexplore.ieee.org/document/8526934

Machine learning applications for building structural design and performance assessment: State-of-the-art review, https://www.sciencedirect.com/science/article/abs/pii/S2352710220334495

State-of-the-art on research and applications of machine learning in the building life cycle, https://www.sciencedirect.com/science/article/abs/pii/S0378778819337879

Pick and Review

I pick *Component-based machine learning for performance prediction in building design* as the reviewed paper.

Introduction

A component-based approach is proposed in this paper to solve the challenge of a sustainable built environment. Different from the traditional building performance simulation, this new method is based on the principles of machine learning. By data analytics, trained models can perform well in different situations.

Generally speaking, physical modeling and simulation and machine learning models are two current main approaches. In terms of modeling categorization, the former is known as white box modeling, and the latter is known as black box modeling. It has already proved that the results of white-box modeling are precise and reliable since they even consider dynamic effects. Nevertheless, they have a high information demand, which means relatively long computation times and costs. By contrast, black box modeling is faster and cheaper. However, due to the monolithic characteristic of the model, internal properties that lead to the prediction are unknown. To better achieve a balance of the computation cost and the number of accessible data, the author presents this component-based approach.

In the component-based approach, the component is a subordinate model of one building part or its technology, such as walls, windows, roof, floor slab, etc. It is defined by input and output parameters. This method uses black box models to connect these parameters and combines them with the paradigms of systems engineering to manage the complexity that arises. In contrast to the monolithic use of surrogate models, it allows the potential of system engineering to be exploited for complexity management. By building models for general components, models can own a much greater degree of generalization. Compared to monolithic black models, it enables designers and engineers to better analyze and understand systemic interdependencies.

Method

To begin with, the approach requires the decomposition and parametrization of the design. Relevant options to be changed in the design require consideration in the performance prediction model. There are two possibilities of varying a design. First, changing parameters of the design, such as the length or width of the building, size of the windows, strength of insulation, etc. Second, changing the structure of the design, such as adding or removing walls, windows, doors, zones, etc. The second type of change plays a pivotal role in designing. It is very difficult to cover these changes using a parametric model, as the combinations of possible changes quickly become unmanageable. Therefore, the author addresses this situation by taking the component-based approach.

The next step is decomposition, which requires the consideration of some criteria to deliver well-suited, suitably generalizing models. In terms of recurrence, the key to generalization is identifying basic reusable elements. The author proposes two different approaches in that respect. First, the breakdown follows building elements as they are normally used in the design, such as walls, windows, doors, roofs, floor slabs, etc. Second, the design is broken down into zones that include the parameters for their enclosure properties. Besides, a suitable parameter structure and a sufficient fit of the surrogate models are also important. Therefore, designers and engineers should make the appropriate choice according to the actual situation.

Third, select the ML modeling strategy. In this paper, the writer uses artificial neural networks (ANN) to represent the components' behavior. This method has a high degree of flexibility when representing data regression. Nonetheless, there might also be other methods for properly representing the component's behavior and response. In their experiments, the ML modeling consists of two types: static and dynamic. The static ML model is used to represent simple phenomena. The dynamic model is developed for a parameterized thermal zone. By applying different models, their method can reduce computation costs as much as possible under the premise of ensuring accuracy.

Then it comes to train data from a parametric simulation. The parametric model sampled 800 design combinations by using a Latin Hypercube sampling scheme. The first part included 400 design combinations that have windows in one orientation only. This was required to cover the effect of windows in each orientation separately to ensure that the data was representative in this respect. The remaining 400 design combinations are based on the variation of all the design parameters. The training data distribution in this sampling is tailored to the expected occurrence of geometric building dimensions in the prediction cases. With this step, an improved training process can be achieved.

The last step is training the ML model for components. For the static response ML model, it used a simple one-layer artificial neural network (ANN) per component. Two algorithms, the Levenberg-Marquardt algorithm and Bayesian regularization, are used for training. The first algorithm was used for the less complex datasets, whereas the latter was used for complex datasets, which are the walls, the windows, and the zone heating response due to interactions between large numbers of parameters. For the dynamic response model, the related components are developed in Python using the Keras library with TensorFlow backend. Adam's training algorithm is used to develop the model. This is a first-order gradient-based optimization algorithm. The data generated is split into training, cross-validation, and test data in a ratio of 70/15/15. Training data is used to develop the ML model, while cross-validation data is used to tune the model structure. Test data is used to evaluate the generalization.

Experiment Results

To prove the reliability of the trained model, results from the Energy Plus physical building simulation software are used to compare with the results from the ML model. On the validation data, most static models achieved high levels of accuracy in terms of R2 close to 0.99, or higher in a test with 200 independent random samples of the design space. A few models got lower. But that is because they have very low response values in the simulation, which causes irregularities to dominate design parameters and makes the component subordinate in performance prediction. As for dynamic models, all of them perform well. Since they cost more during the train, it is reasonable to get more accurate results.

The author also chooses some whole building test cases to prove the practicability of their method at a building level. For the first two simple cases: a two-storey rectangular building and an eight-storey box building, their models all perform well. He admits that it is because the test case partly falls under the training distribution. As for the last case, a more complex design, results become lower but are still acceptable. The author explains that there exists several new data. After models get trained with these data, they can still predict results similar to reality. He also believes that this slight increase is a positive indication of the extensibility of the component-based modeling approach.

Future Work

A better choice can be made in the selection of static ML model and dynamic ML model. As the author mentioned before, all dynamic models work perfectly while some static models can be improved. This is a trade-off between the computation cost and prediction accuracy. Therefore, it is possible to find and change some less important components from dynamic models to static models to save money. At the same time, some factors, which we think were unimportant before the experiment, may need to be changed from static models to dynamic models for a more reliable prediction.

Another potential improvement is adding more components. One fundamental limitation of ML is the training data range. The ML model is only valid for cases that are covered by the training data. By adding more components, the data range is increased, thus the machine after training will be able to make reliable predictions in more complex designs. Models get a much greater degree of generalization.

A further aspect making the component-based approach valuable is improving model training methods. With machine learning, the basic limitation is that models can only work on data that they trained before. There may exist some factors that affect buildings in reality but designers fail to find and consider them when training models. In that case, models are never able to make correct predictions. However, if more advanced technology is applied, such as replacing machine learning with deep learning, models will get the ability to self-study. Even if the machine only gets several components from designers and engineers at the beginning, it finds and adds extra components by itself during the training process. Finally, reliable and precise models can be produced.

Target User

One main target user is the architect. The first impact is reflected in the reliability of the product. If the product fails to always predict accurately, no architect would like to use it. Once the machine gives wrong results, the architect applies them to his designs and then later the building accident happens due to this mistake, both the architect's life and the resident's life break. Therefore, architects will not take the risk of using this product. Another factor affected by the target user is the computation cost. As mentioned earlier, it is important to find an appropriate trade-off between the computation cost and the prediction accuracy. If the price is so expensive that architects will recover their costs after finishing hundreds of orders, nobody will choose it.