Applied Energy 240 (2019) 35–45

Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy

Deep learning-based feature engineering methods for improved building energy prediction

Cheng Fan[a], Yongjun Sun[b], Yang Zhao[c,*], Mengjie Song[d], Jiayuan Wang[a]

[a] Department of Construction Management and Real Estate, Shenzhen University, Shenzhen, China
[b] Division of Building Science and Technology, City University of Hong Kong, Hong Kong, China
[c] Institute of Refrigeration and Cryogenics, Zhejiang University, Hangzhou, China
[d] Department of Human and Engineered Environmental Studies, Graduate School of Frontier Sciences, The University of Tokyo, Japan

HIGHLIGHTS

- Deep learning techniques are utilized to facilitate building energy predictions.
- Unsupervised autoencoders are developed for feature engineering.
- Generative adversarial networks are utilized for feature engineering.
- Generative modeling proves to be useful for enhancing prediction performance.
- This study enables an automated approach for building energy modeling.

ARTICLE INFO

Keywords:
Building energy prediction
Data mining
Intelligent buildings
Unsupervised deep learning
Generative adversarial networks

ABSTRACT

The enrichment in building operation data has enabled the development of advanced data-driven methods for building energy predictions. Existing studies mainly focused on the utilization of supervised learning techniques for model development, while overlooking the significance of feature engineering. Feature engineering are helpful for reducing data dimensionality, decreasing prediction model complexity, and tackling the problem of corrupted and noisy information. Considering that each building has unique operating characteristics, it is neither practical nor efficient to manually identify features for model developments. Data-driven feature engineering methods are thus needed to ensure the flexibility and generalization of building energy prediction models. Using operation data of real buildings, this paper investigates the performance of different deep learning techniques in automatically deriving high-quality features for building energy predictions. Three types of deep learning-based features are developed using fully-connected autoencoders, convolutional autoencoders and generative adversarial networks respectively. Their potentials in building energy predictions have been exploited and compared with conventional feature engineering methods. The study validates the usefulness of deep learning in enhancing building energy prediction performance. The research results help to automate and improve the predictive modeling process while bridging the knowledge gaps between deep learning and building professionals.

1. Introduction

Building operations are energy intensive, accounting for approximately 80–90% of the total energy consumption throughout the whole building life-cycle [1]. The enhancement in building operational performance can provide key solutions to building energy savings due to the wide existence of manual faults, operating deficiencies, improper control strategies and etc. [2,3]. Meanwhile, building operations are information intensive thanks to the adoption of intelligent building management systems. Massive amounts of building operation data are being recorded and available for data analysis. It is very promising to develop big data-driven approaches to smart building energy management.

Among many data analytics, predictive modeling has drawn great attentions from both academic researchers and building professionals. The common prediction targets in the building field include building energy consumptions [3,4], indoor environment [5] and system performance indices [6]. Predictive modeling is closely related to two main

* Corresponding author.
E-mail address: youngzhao@zju.edu.cn (Y. Zhao).

https://doi.org/10.1016/j.apenergy.2019.02.052
Received 27 September 2018; Received in revised form 25 January 2019; Accepted 7 February 2019
Available online 13 February 2019
0306-2619/

tasks in building energy management, i.e., optimal controls and anomaly detections. As examples, the accurate and reliable predictions of building cooling load can be used for optimizing the operation strategy of building thermal energy storage systems [7], finding the most energy-efficient setpoints of chilled water systems [8,9], and deriving the optimal on–off schedules for chiller plants [10]. Anomalies or faults can also be detected through data-driven approaches. In such a case, a prediction model is developed to represent the normal operation patterns in historical data, and anomalies or faulty conditions can be detected by comparing with the predicted or actual values. Such approaches have been used to detect anomalies at multiple levels, such as abnormal energy profiles at the building-level [11] and faulty operations at HVAC system-level [12].

In general, existing methods on building energy predictions can be divided into two general groups, i.e., physical and data-driven methods [13]. Physical methods mainly rely on physical principles and domain expertise to specify the relationships between model inputs and outputs. These models are typically referred as "white-box" models. The main limitation is that the "white-box" modeling process can be time-consuming as excessive details are needed, such as the properties of building envelopes and theoretical operating characteristics of service systems. By contrast, data-driven models aim to discover the underlying relationships between input and output variables based on actual operation data. The modeling process is typically more efficient and flexible for practical applications [13]. Considering the nonlinear nature between building energy consumptions and its affecting variables, advanced supervised learning techniques have been used for model developments. The most widely used supervised learning techniques in the building field include artificial neural networks, support vector regression and ensemble methods [14,15]. Compared with statistical methods (e.g., multiple linear regression), these techniques can describe more complicated and nonlinear relationships and thus, the predictions obtained are typically more accurate.

Despite encouraging results obtained, most studies in this area focused on the utilization of supervised learning techniques in model developments, while overlooking the other key factor in predictive modeling, i.e., feature engineering. Feature engineering refers to the process of constructing valuable features as model inputs. Existing studies mainly used engineering expertise to select inputs for developed models. For instance, building cooling loads are closely related to indoor occupancy [16]. Considering that the building occupancy is correlated with time, time variables (e.g., *Hour* and *Day type*) are usually selected as model inputs [17,18]. The outdoor environment can dramatically affect the building cooling load. Therefore, variables which can describe outdoor conditions, such as outdoor dry-bulb temperature and relative humidity, are also used as model inputs [19–21]. Similarly, variables which are closely related to chiller operating performance (e.g., the chilled water supply and return temperature) were used as inputs for estimating chiller power consumption [22,23]. Considering that building operations are dynamic and autocorrelated, more reliable and accurate predictions can be achieved by introducing variables describing operating conditions at previous time steps, e.g., using the building energy consumptions in previous hours as inputs [24,25]. It should be noted that introducing extra variables on previous timesteps may negatively affect the prediction performance. The main reasons are as below. Firstly, these variables are highly correlated and therefore, the supervised algorithm may not be able to identify the true relationships across different time steps. Secondly, introducing too many input variables will dramatically increase the model complexity. It will typically increase the risk of over-fitting while enlarging the computational burdens.

To tackle this problem, previous studies mainly adopted two methods to construct features from historical building operation data. The first relies on domain expertise to manually select certain historical values as model inputs [26]. Such methods cannot serve as generic solutions and the generalization performance is typically poor. The other adopts statistical methods to extract data-driven features. One popular method is to calculate the summarizing statistics (e.g., the mean and standard deviation) of measurements over a time period [27]. It can effectively reduce the number of input variables. However, the information loss may be significant when the window size is large and time series data are highly fluctuating. The other popular statistical feature engineering method is principal component analysis. The principal components obtained can be used as model inputs [28,29]. The information loss can be controlled based on the total variance explained by principal components. It is a more advanced statistical approach to feature engineering and can be used to tackle the problem of multicollinearity among input variables. Nevertheless, the features developed are in essence linear transformations of the original data. Consequently, their value is limited in representing high-level interactions in the original data. To ensure the reliability and flexibility in building energy predictions, it is desired to develop advanced data-driven feature engineering methods to construct nonlinear, high-level and useful features for modeling. Such methods can help to fully automate the predictive modeling process while ensuring better generalization performance.

Deep learning is a powerful technology which has been used in a wide variety of analytical tasks, such as image classification and speech recognition [30,31]. Compared with conventional data analytics, deep learning models have deeper architectures, i.e., the input data are transformed multiple times before deriving the output. It can therefore describe complicated data relationships. Deep learning can be used either in a supervised or unsupervised manner. Unsupervised deep learning has been proved to be useful in constructing high-level features from image and audio data. However, few studies have been performed to investigate the potentials of unsupervised deep learning in the building field. The basic versions of unsupervised deep learning models were used to construct features for building energy modeling [3,11,32]. The results showed that the features derived were capable of preserving useful information in the original data, based on which accurate predictions could be achieved. It is noted that powerful deep learning techniques are constantly emerging, providing promising techniques for analyzing complicated time series data in the building field. To fill the knowledge gap between deep learning and building professionals, this study examines the potential of various state-of-the-art deep learning techniques for feature engineering, i.e., fully-connected autoencoders (AEs), convolutional autoencoders (CAEs) and generative adversarial networks (GANs). Using field data, the usefulness of deep learning-based feature engineering methods is assessed and compared with conventional feature engineering methods. It aims to provide useful references for the automatic developments of accurate and reliable building energy prediction models.

2. Research methodology

2.1. Outline

As shown in Fig. 1, the research methodology consists of two main parts, i.e., feature engineering and predictive modeling. The first step performs data-driven feature extraction based on different feature engineering methods. In total, five feature engineering methods are adopted, including two conventional data-driven feature engineering methods and three deep-learning based methods. Secondly, four supervised learning techniques, including multiple linear regression, support vector regression, artificial neural networks and extreme gradient boosting trees, are used to develop one-step ahead building energy prediction models based on different feature sets. The resulting prediction accuracies are used to evaluate the usefulness of different feature engineering performance.

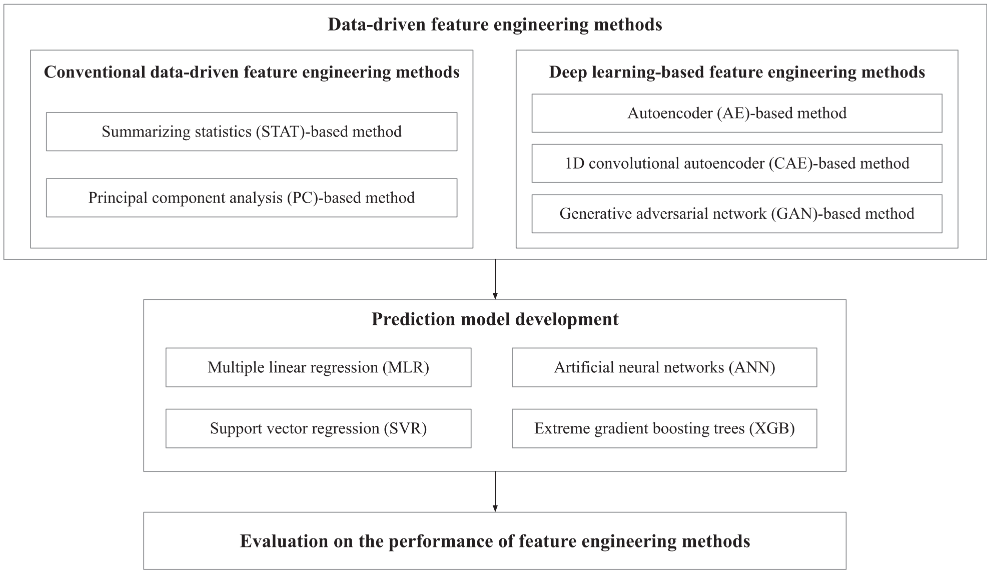

Fig. 1. Research outline.

2.2. Data-driven feature engineering methods

In this study, the performance of five data-driven feature engineering methods is assessed and compared. The first two are conventional data-driven feature engineering methods, which rely on principal component analysis and simple summarizing statistics. Three deep learning-based methods are proposed for feature engineering based on the use of fully-connected autoencoders (AE), one-dimensional convolutional autoencoders (CAE) and generative adversarial networks (GAN) respectively. To fairly evaluate the performance of different feature engineering methods, the number of features extracted is fixed as k and determined as the number of principal components selected for predictive modeling. It should be noted that determining the number of principal components to retain is one of the most critical challenges in principal component analysis. One of the most widely used approach is to select principal components based on the proportion of total variance explained [33]. In practice, the proportions used typically range between 70% and 95% [34]. In this study, a rather conservative value, i.e., 95%, is adopted to minimize the information loss. As a result, the first feature set for model development consists of k principal components. The second feature set contains k summarizing statistics, i.e., the building cooling load measurements in previous hours are equally divided into k segments and their mean values are calculated as features. The third to fifth feature sets are constructed by deep learning-based feature engineering methods. The architectures of deep learning models are carefully designed to ensure that the number of features is k. The details of these three methods are shown as follows.

2.2.1. The autoencoder-based feature engineering method

An autoencoder (AE) can be regarded as an artificial neural network trying to reconstruct the input data. It has two parts, i.e., the encoder and the decoder. Assuming the input data is X with p variables, the encoder tries to learn a function f which encodes the input data X into a hidden space h, i.e., $h = f(X)$. Meanwhile, the decoder learns a function g to reconstruct X as close as possible based on h, i.e., $X' = g(h)$ so that $X \approx X'$. The development of AEs belongs to the category of unsupervised learning, as the model outputs are simply set as the model inputs. In general, there are two basic feed-forward AE architectures, i.e., undercomplete and overcomplete feed-forward AEs. As shown in Fig. 2(a), an undercomplete feed-forward AE adopts a bottle-neck architecture for input reconstruction, i.e., the dimension of h is smaller than p. By contrast, as shown in Fig. 2(b), the dimension of h in an overcomplete feed-forward AE is larger than p. Given too much capacity, an AE may fail to learn any meaningful functions for encoding and decoding, as a trivial identity function may be learnt for input reconstruction when h is equal or larger than p. Therefore, regularization techniques, such as representation sparsity and robustness to noises, are usually used during the training process. AEs have gained great popularity for dimensionality reduction and information retrieval tasks [35,36]. Given that an autoencoder has learnt meaningful functions for input reconstruction, the output of the encoder, i.e., h, can be utilized for input representation, based on which better supervised learning performance can be achieved. As indicated by Goodfellow et al., lower-dimensional representations can enhance the performance of regression and classification tasks [37]. Therefore, in this study, a bottle-neck architecture is adopted for developing basic AEs. The outputs of the encoder are utilized as features for building energy prediction models.

2.2.2. The convolutional autoencoder-based feature engineering method

Traditional AEs adopt a feed-forward architecture for input reconstruction, i.e., each output unit is linked with each input unit. As a result, the model complexity can be very high due to the large number of model parameters. More importantly, the feed-forward architecture cannot accurately capture the dependency in data with a grid-like topology, e.g., one-dimensional time series data and two-dimensional image data [38]. A more efficient model architecture, i.e., the convolutional neural network, can be used to tackle this challenge. Convolutional neural networks adopt convolutions as the basic operation unit, which results in two intrinsic model characteristics, i.e., sparse interactions and parameter sharing [37]. Sparse interactions refer to the way of kernel computations. Traditional AEs utilizes a matrix of parameters to describe the interaction between each input and output units. Assuming the input and output data have p and q variables respectively, the resulting parameter matrix has a size of $p \times q$. By contrast, convolutional neural networks adopt a set of kernels with a small size of c to describe the interactions with each output unit. Consequently, the parameter size becomes much smaller, i.e., $c \times q$ while $c \ll p$. In addition, in the traditional feed-forward architecture, each model parameter is used once when calculating the output. For convolutional neural networks, the model parameters are shared across different operations, which can further reduce the computational burdens.

Convolutional neural networks have been widely used in analyzing time series data and image data. It has presented great capability in extracting temporal or spatial features, such as recognizing shapes and

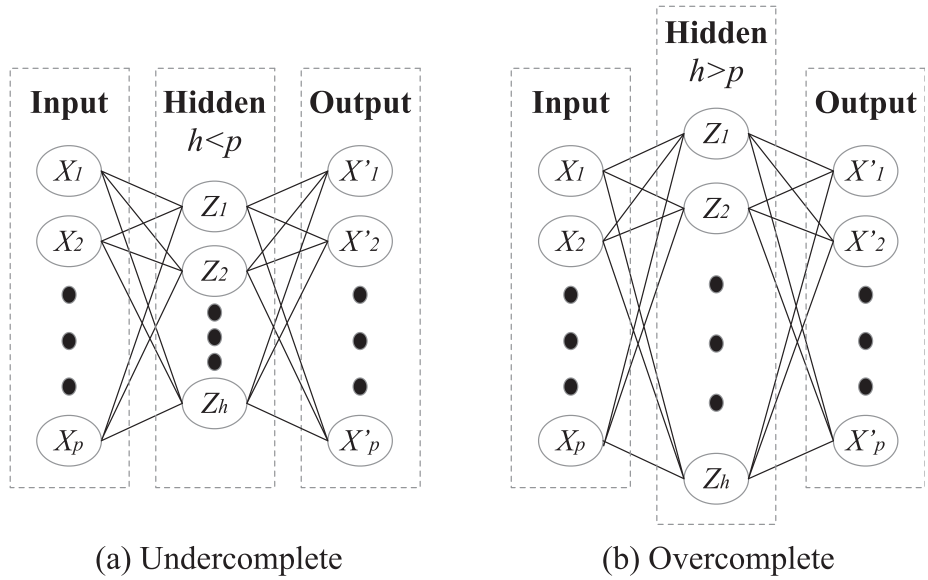

Fig. 2. Basic architectures of feed-forward autoencoders.

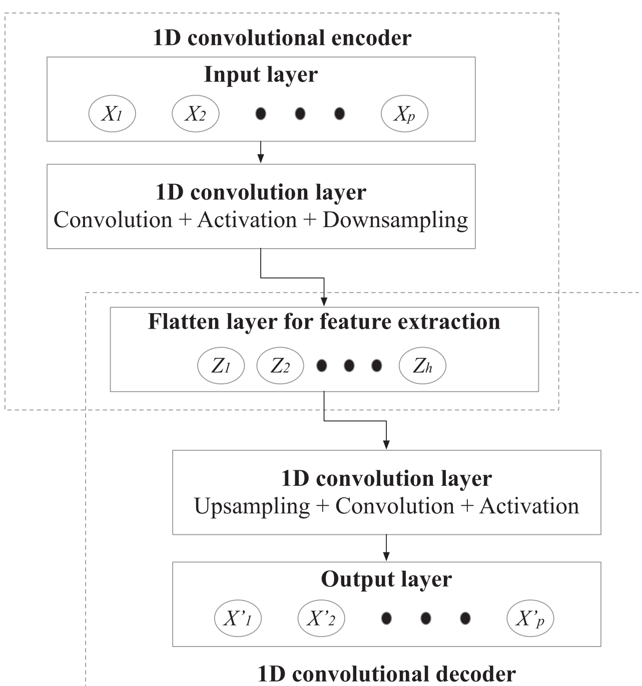

Fig. 3. Basic architecture of one-dimensional convolutional autoencoders.

edges in image data [39]. Considering that building energy data are essentially time series data, the adoption of one-dimensional convolutional autoencoders (CAEs) may capture more useful features to describe the underlying temporal relationships for predictive modeling than the fully-connected AEs. Fig. 3 presents an example of one-dimensional CAE, where convolution units are used for information encoding and decoding. In this study, the outputs of encoder are utilized as features for developing building energy prediction models.

2.2.3. The generative adversarial network-based feature engineering method

As one of the most promising techniques in the field of deep learning, generative adversarial networks (GANs) have attracted worldwide interests for the task of generative modeling. The concept of GAN was firstly proposed by Goodfellow et al. in 2014 [40]. As shown in Fig. 4, the learning framework of GANs contains two models, i.e., a generator G and a discriminator D. The generator G aims to learn a distribution P_{model} which mimics the real data distribution P_{data}, while the discriminator G tries to distinguish whether a sample is drawn from P_{data} or P_{model}. The input data to G are usually noises drawn from a random distribution, such as uniform or normal distributions. Assuming the parameters of G and D are θ_G and θ_D respectively, the learning process of GAN can be summarized as follows. Firstly, a set of random noises (i.e., denoted as z) are drawn from a random distribution and fed to the generator G to produce synthetic samples, i.e., $G(z|\theta_G)$. Secondly, synthetic samples are combined with real samples as input for the discriminator D. The discriminator D is trained to identify whether a sample is synthetic or real, while the generator G tries to fool the discriminator D by generating high-quality synthetic samples that are very close to real ones. The adversarial game played by G and D can be expressed in the following objective function: $\min_G \max_D V(D, G) = E_{x \sim P_{data}} \log[D(x)] + E_{x \sim P_{model}} \log[1 - D(x)]$, where $D(x)$ is the discriminator loss in classifying data samples, $x \sim P_{data}$ and $x \sim P_{model}$ indicate whether the input to the discriminator comes from the real data samples or the generator model. GANs have been utilized for various applications, such as image-to-image translation and text-to-image translation [41]. Despite its power in generative modeling, GANs are notorious for their training difficulties. To overcome this problem, many techniques and guidelines have been proposed, such as adopting batch normalization, dropouts and rectified linear units during the training process [42,43].

GAN models provide an alternative approach to deriving meaningful features for supervised learning. If the GAN learning process converges, the generator G should be able to generate indistinguishable samples from real data, while the discriminator D can capture the key characteristics for classifying real and synthetic samples. In such a case, the activations before the discriminator D's output layer must convey useful information for supervised learning. Similar approaches have been used in the field of computer vision [43,44]. The research results showed that GANs could learn good representations from the original data, based on which better image classification performance was achieved. In this study, the same idea is implemented to extract features for developing building energy prediction models. To the best of the authors' knowledge, this is the first attempt in the building field to exploit the power of GANs in analyzing building energy data. As demonstrated in the following sections, GANs are not only helpful for deriving features for supervised learning, but also generating realistic profiles on building energy consumptions. Such building energy profiles

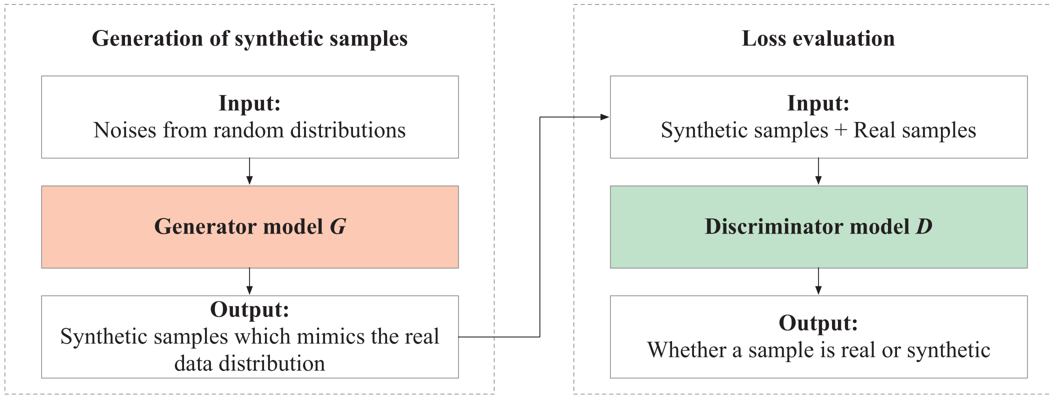

Fig. 4. The learning framework of generative adversarial networks (GAN).

could be valuable for performance benchmarking, yet this is not the focus of this study.

2.3. Supervised learning techniques for building energy predictions

Once different sets of features are extracted, supervised learning techniques are used to build prediction models. To comprehensively examine the usefulness of features extracted, four representative techniques, which differs in their nonlinearity handling abilities, model architectures and inference mechanisms, are selected, i.e., the multiple linear regression (MLR), artificial neural networks (ANN), support vector regression (SVR) and extreme gradient boosting trees (XGB). The MLR serves as the performance benchmark as it can only describe linear interactions between input and output variables. The other three are capable of capturing nonlinear interactions. The SVR and ANN are selected due to their wide adoption in the building energy prediction field [45]. The XGB is a more advanced ensemble learning technique [46]. It is developed based on the concept of boosting trees, which has proved to be very powerful in modeling complicated relationships [47]. The adoption of these four supervised learning techniques can help to answer a variety of problems related to the topic concerned, e.g., which feature engineering method provides the best prediction performance in terms of different modeling techniques.

3. Case study

3.1. Building and data descriptions

The building operation data retrieved from an educational building in Hong Kong were used for analysis. The building mainly contains classrooms for students, offices for university staffs, and a data center for computing devices. It has a gross floor area of 11,000 m^2, out of which approximately 80% of areas are air-conditioned. The building is served by a complicated central air-conditioning system, including four water-cooled chillers (i.e., three with a cooling capacity of 1932 kW each and one with a cooling capability of 540 kW) and four cooling towers. Six constant-speed primary and six variable-speed secondary pumps are used for chilled water circulation, while six constant-speed water pumps are used for distributing condensing waters.

The operation data were collected with a sampling interval of 30 min. The whole year data in 2015 were adopted for analysis. In total, the data have 17,040 observations. The variables can be divided into three general categories: (1) time variables (*Month*, *Day*, *Hour*, *Minute* and *Day type*); (2) outdoor variables (outdoor dry-bulb temperature and relative humidity); (3) operating parameters of the chiller plant (e.g., the temperatures and flow-rates of chilled water and condenser water). The total building cooling loads were calculated based on the chilled water flow rate, the supply and return chilled water temperature. The whole data were divided into training and testing data sets with proportions of 70% and 30% respectively. Standardization and one-hot encoding were used to preprocess numerical and categorical input variables respectively. The feature engineering and prediction model development are performed based on the training data set, while the prediction performance was calculated and reported based on the testing data set.

3.2. Construction of feature sets

Prediction models are developed to forecast the one-step ahead building cooling load based on two types of inputs: (1) external features, which refers to the time variables (e.g., *Hour* and *Day type*) and outdoor environment variables (e.g., outdoor dry-bulb temperature and relative humidity) at time $T + 1$. These variables are used to reflect the influence of indoor occupancy and outdoor environment on building energy. It should be noted that other environmental variables, such as solar radiations and wind speed, may also influence the building cooling load. However, these variables are typically not available in practice and therefore, are neglected in this study; (2) internal features, which are derived from the historical measurements of building cooling loads from T-m to T, where m is the maximal time lag considered for feature engineering. The parametric spectral density estimation method was used to identify the intrinsic periodicities in building cooling load. As shown in Fig. 5, the most significant frequency identified is 0.021, which corresponds to a period of 48 time steps (i.e., $\frac{1}{0.021}$). Since data samples were collected with a time interval of 30 min, it indicates a significant daily periodicity. Therefore, the maximal time lag m considered for feature engineering was set as 48, i.e., 24 h.

The feature engineering methods described in Section 2.2 were used for feature extraction. Fig. 6 presents the cumulative variance explained by the first twenty principal components. It is shown that the first six

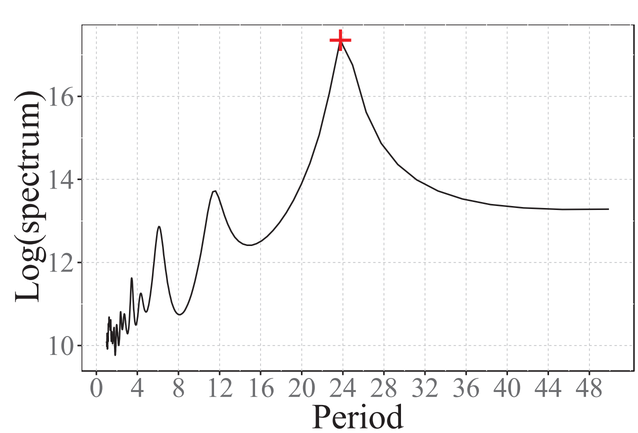

Fig. 5. Spectrum density estimation for the time series of building cooling load.

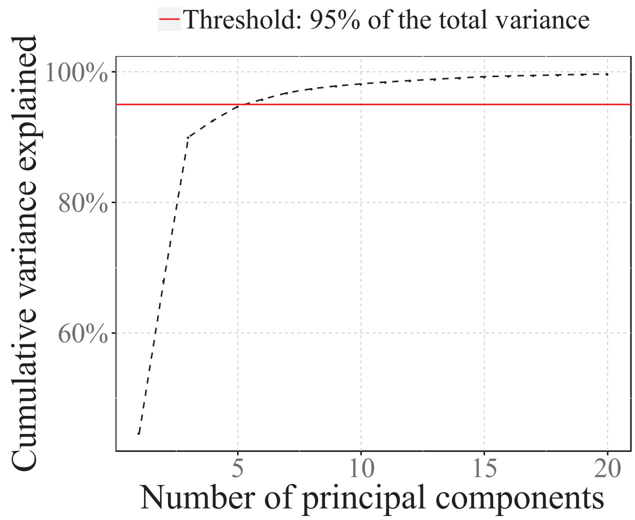

Fig. 6. The cumulative variance explained by principal components.

principal components are capable of explaining 95% of the total variance. Furthermore, the gain in total variance explained is rather small when including extra principal components. Therefore, the number of features k used for predictive modeling was set as six. The first feature set consists of the first six principal components. The second feature set was constructed by calculating the mean values of six equally divided temporal segments in the past 24 h.

The third feature set was extracted based on a feed-forward autoencoder (AE). The architecture and techniques used are illustrated in Fig. 7. The input and output layers both have 48 neurons, representing the original building cooling load measurements in previous 24 h. The model has three hidden layers and optimization was performed to fine-tune the parameters, such as the dropout at different layers, the activation function used and whether applying batch normalization or not. The number of neurons at the middle layer was set as six, corresponding to the feature number k used in previous statistical methods. The fourth feature set was derived based on the one-dimensional convolutional autoencoder (CAE). The architecture and techniques used are illustrated in Fig. 8. It was designed in such a manner that the number of neurons at the encoder's output layer is six.

Fig. 9 presents the generative adversarial networks (GAN) developed in this study. To ensure the overall performance, both generator G and discriminator D adopt the one-dimensional CAE as the basic operation unit. The leaky rectified linear unit is adopted as the activation function in hidden layers. The slope was set as 0.3 as recommended in [48]. The input to G is sampled from a Gaussian normal distribution with a mean of zero and a standard deviation of one. The input dimension was set as ten, which means that each synthetic sample on the

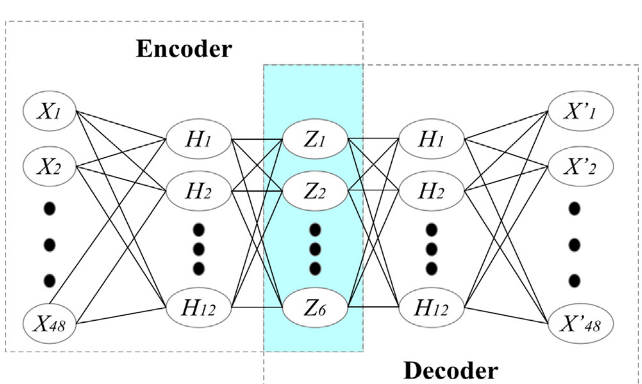

Fig. 7. The schematic of AE used for feature engineering.

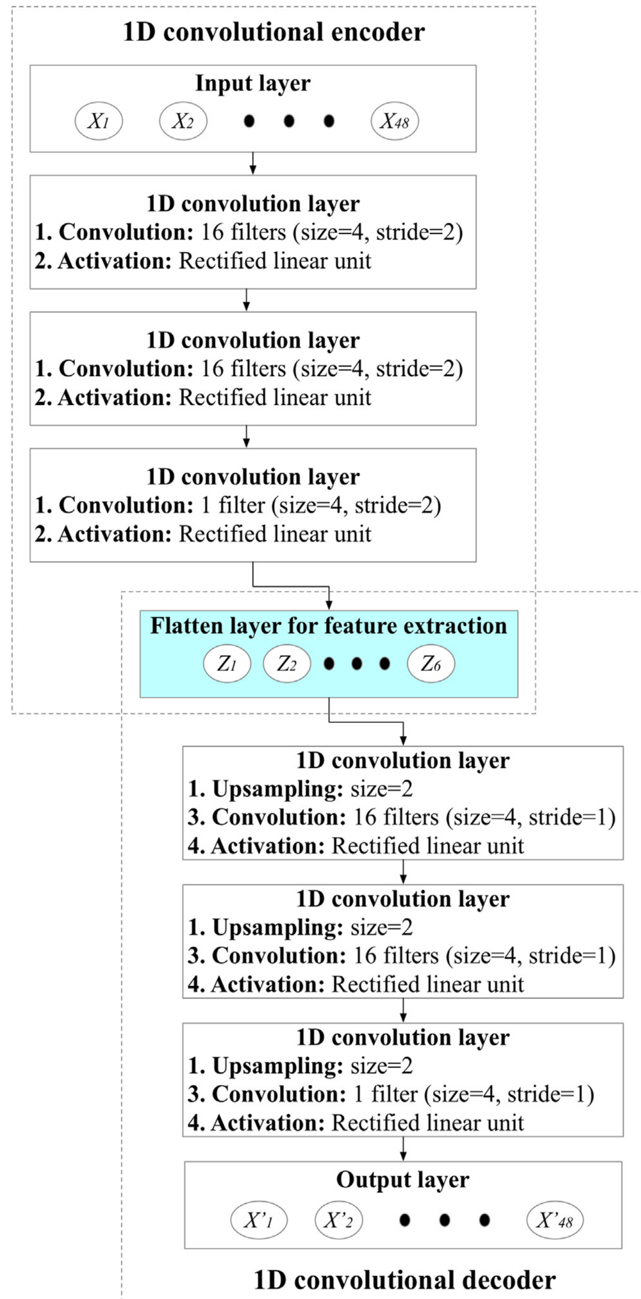

Fig. 8. The schematic of CAE used for feature engineering.

past 24 h building cooling load is constructed based on ten random Gaussian noises. The discriminator D aims to distinguish whether a sample is real or synthetic and therefore, the output layer has only one neuron and the *sigmoid* function (i.e., $\frac{1}{1+e^{-z}}$) was used for classification. A fully-connected layer, which has six neurons, was designed before the output layer for feature extraction. It should be mentioned that GAN models are quite difficult to train and there is no easy way to quantitatively evaluate the GAN performance. In this study, the performance of GAN is assessed from two perspectives: (1) whether the losses of G and D are behaving normally, i.e., neither G nor D dominates the learning process and they are actually learning towards equilibrium; (2) whether the generated synthetic samples are close to real 24 h building cooling load profiles or not. If so, it indicates that G has learnt how to generate realistic samples and therefore, D has learnt how to distinguish real and synthetic samples. As examples, Figs. 10 and 11 present examples of synthetic 24 h cooling load profiles generated by G and real

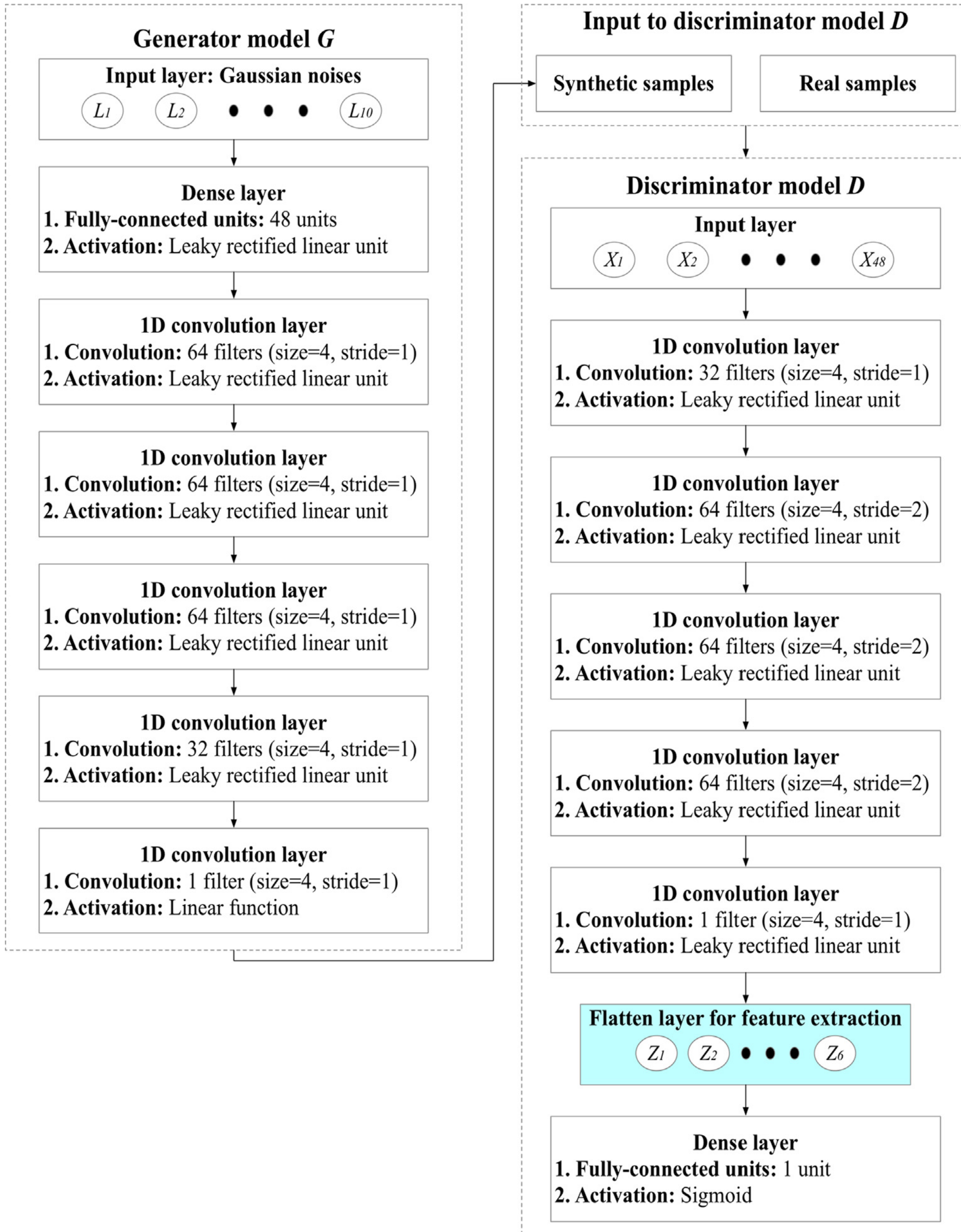

Fig. 9. The schematic of GAN used for feature engineering.

cooling load profiles respectively. Considering that the synthetic profiles generated are very close to real ones, the features extracted by *D* should contain useful information about the intrinsic data characteristics for predictive modeling.

3.3. Development of building cooling load prediction models

Given a feature set and a supervised learning algorithm, the prediction model parameters were optimized through five-fold cross-validation. The generalization performance was obtained by applying the optimized prediction models to the testing data sets. The resulting accuracies were used as indicators to reflect the quality of different feature sets.

More specifically, the Gaussian radial basis kernel was used for SVR model development. Two key parameters, the complexity parameter *C* and the smoothing parameter *sigma*, were optimized to ensure the model performance while avoiding the problem of over-fitting. A larger *C* leads to a more complicated model while a larger *sigma* leads a more smooth and flexible decision boundary. The candidate values of these two parameters take a form of 2^x, where × are integers ranging from −10 to 10. A three-layer multi-perceptron architecture was used for developing ANN models. Optimization was performed to select the most suitable activation functions for the input and hidden layers, i.e., *sigmoid*, *Tanh* and *ReLU* (i.e., rectified linear units). The number of hidden neurons was set as 3 based on one of the rules of thumbs in neural network design, i.e., $\frac{No.\,of inputs + No.\,of outputs}{2}$ [49]. XGB models can be regarded as a decision tree-based ensembling method. In this study, the number of trees was set as 500. Optimization was performed over the tree depth and learning rate. The tree depth specifies the maximal depth of each individual decision tree. The candidate values were set from two to five. The learning rate quantifies how quickly a tree model adapts to the errors in pervious iteration. It was set between 0.0025 and 0.05 with a decimal increment of 0.0025. In general, a smaller learning rate helps to avoid the problem of overshooting, yet it may require more iterations and computation time for convergence.

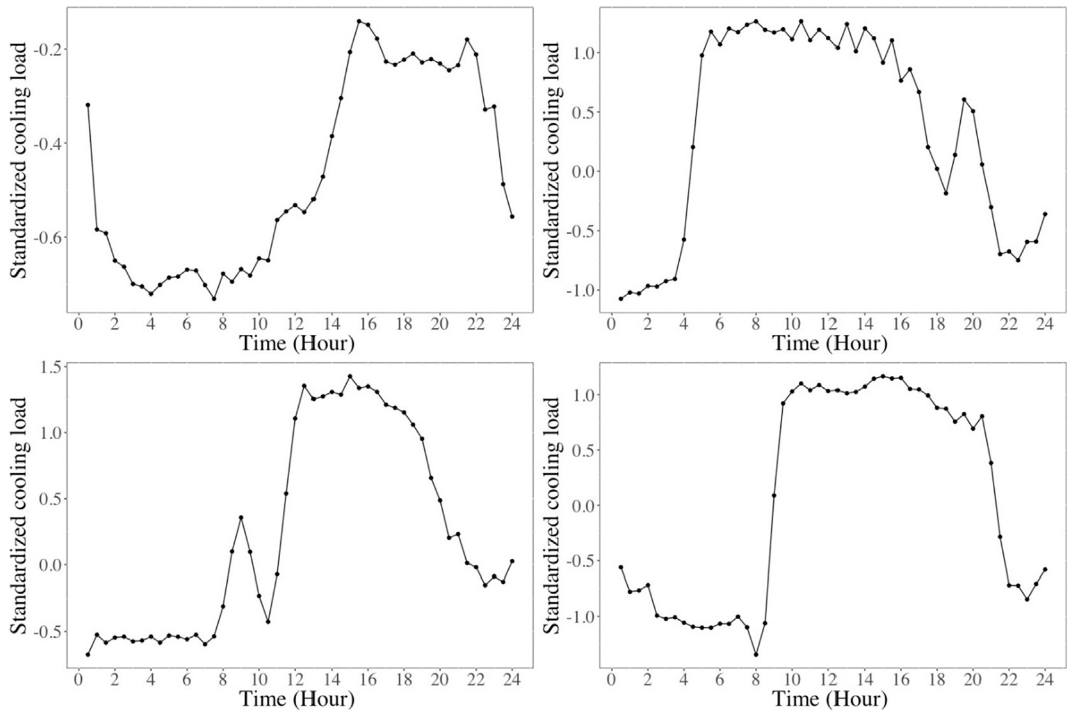

Fig. 10. Synthetic samples generated on 24 h building cooling load.

4. Results and discussions

All the computation works were performed based on the *R* programming language and the deep learning models were realized based on the *Keras* package [48,50]. Three accuracy metrics, i.e., the root mean squared error (*RMSE*), the mean absolute error (*MAE*) the coefficient of variation of the root mean squared error (*CV-RMSE*), were used as performance indicators. The first two are scale-dependent metrics, which can describe errors in their original scales. The latter is a scale-independent metric, which is more suitable for performance comparison across different data sets. The formulas used are shown in Equations-1 to 3 respectively, where y and $\hat{y}$ are the actual and predicted values, and n is the total number of observations.

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

$$MAE = \frac{\sum_{i=1}^{n} |\hat{y}_i - y_i|}{n} \tag{2}$$

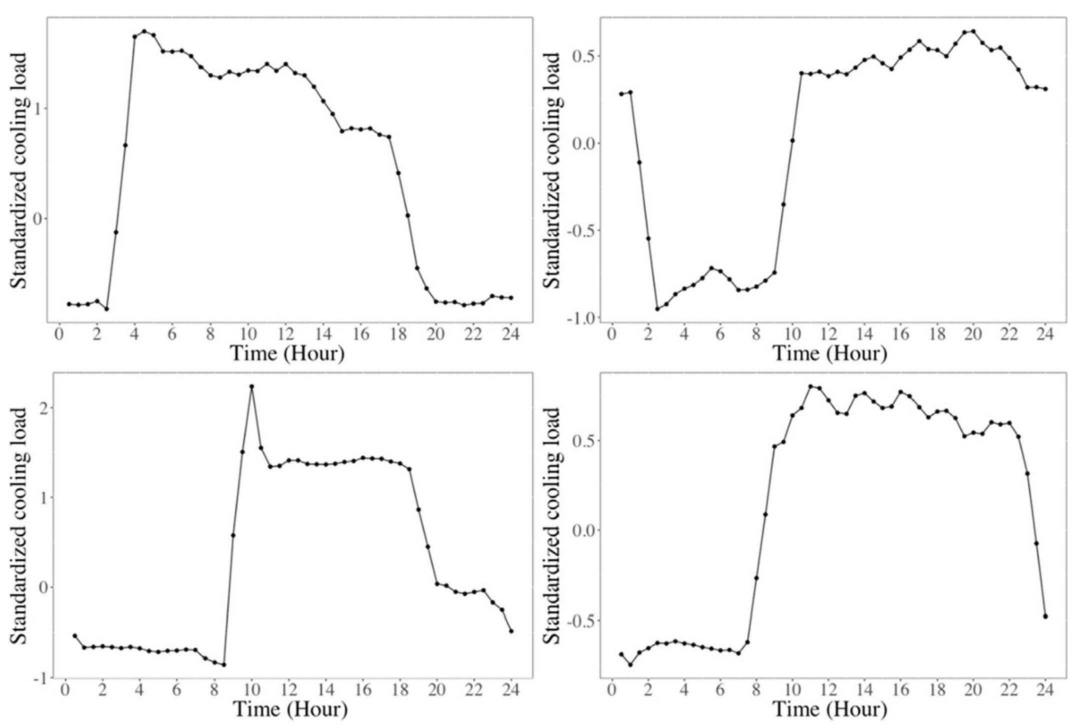

Fig. 11. Real samples on 24 h building cooling load.

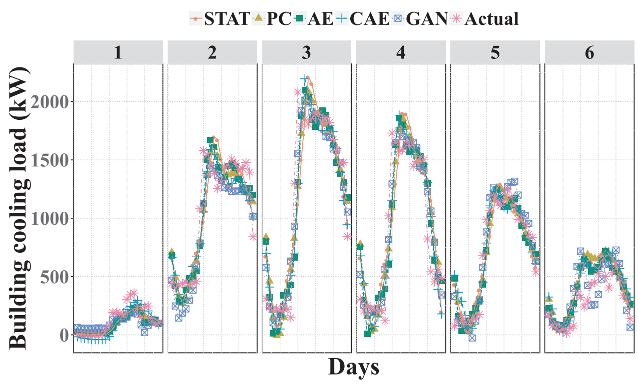

Fig. 12. Prediction performance based on different feature sets and MLR.

$$CV-RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}} \Bigg/ \frac{\sum_{i=1}^{n} y_i}{n} \tag{3}$$

4.1. Prediction performance in terms of supervised learning techniques

Figs. 12–15 depict the one-step ahead prediction performance given different feature sets and supervised learning techniques. The resulting accuracy metrics are reported in Table 1.. It should be mentioned that performance degradation is typically expected when applied for multi-step ahead prediction. The same data set was used to investigate the performance of different strategies for 24 h ahead building cooling predictions [51]. The results indicated that prediction errors generally enlarge 2 to 2.5 times along the 24 h prediction horizon, i.e., if the CV-RMSE at time step $T + 1$ is 10%, then the *CV-RMSE* may range from 20% to 25% at time step $T + 48$ (considering a 30 min time interval) [51]. As indicated in [52,53], a model is applicable for engineering purposes if the *CV-RMSE* is below 30% when using hourly data. It is shown that all the nonlinear modeling techniques are capable of generating predictions with *CV-RMSE* less than 30%. By contrast, the benchmarking modeling technique, i.e., multiple linear regression, can only generate fairly accurate predictions using the GAN-based features. The *CV-RMSEs* calculated for SVR and ANN models are quite close, indicating that these two supervised learning techniques have rather similar prediction power for building energy predictions. Given the same feature set, the XGB models always generate the best prediction performance. It indicates that compared with single model-based predictions, the ensembling methods, which rely on a number of base models to generate final predictions, are more accurate and reliable. It should be noted that the supervised learning techniques used in this study are only representatives of available techniques. Other techniques, such as random forests and time series models, can also make high-quality predictions on building energy consumptions [45]. In practice, the main consideration for supervised learning technique

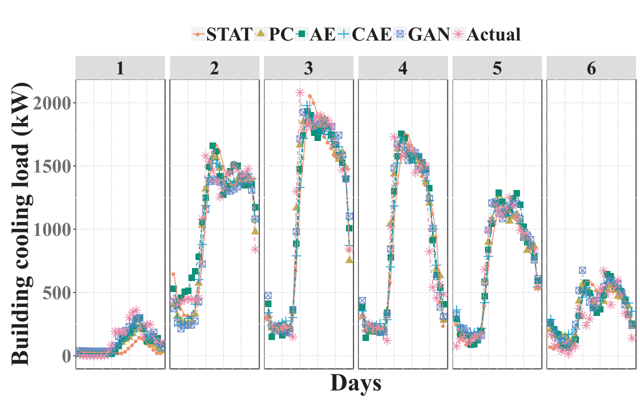

Fig. 13. Prediction performance based on different feature sets and ANN.

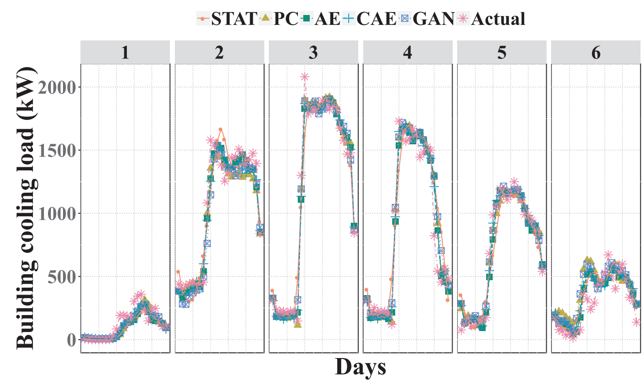

Fig. 14. Prediction performance based on different feature sets and SVR.

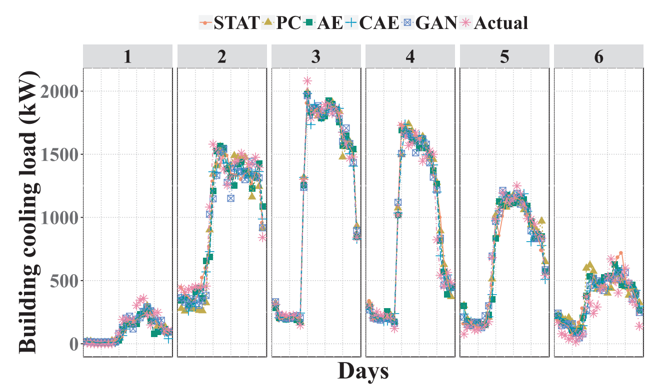

Fig. 15. Prediction performance based on different feature sets and XGB.

Table 1
Prediction performance on testing data set.

Features	Metrics	MLR	SVR	ANN	XGB
PC-based features	RMSE (kW)	209.2	137.1	159.3	122.4
	MAE (kW)	137.9	82.6	96.2	76.2
	CV-RMSE	32.9%	21.6%	25.1%	19.3%
STAT-based features	RMSE (kW)	217.8	160.4	158.2	122.8
	MAE (kW)	142.4	93.0	92.3	74.5
	CV-RMSE	34.3%	25.3%	24.9%	19.3%
AE-based features	RMSE (kW)	191.6	128.1	149.1	120.7
	MAE (kW)	125.8	75.0	90.4	73.6
	CV-RMSE	30.2%	20.2%	23.5%	19.0%
CAE-based features	RMSE (kW)	191.2	131.2	143.0	116.2
	MAE (kW)	127.3	77.9	85.6	72.1
	CV-RMSE	30.1%	20.7%	22.5%	18.3%
GAN-based features	RMSE (kW)	167.9	135.3	127.5	112.6
	MAE (kW)	117.1	79.6	82.1	71.1
	CV-RMSE	26.4%	21.3%	20.1%	17.7%

selection is the balance between prediction accuracy and model interpretability. Linear models are generally more interpretable, yet the prediction accuracy may not be satisfactory. Meanwhile, nonlinear techniques, especially ensembles of nonlinear models, can be very accurate at the cost of model interpretability. Therefore, a promising research direction is to develop post-mining methods to enhance the interpretability of complicated prediction models [54,55]. The insights obtained can help building professionals to better integrate domain expertise with data-driven inference mechanism learnt for decision-makings.

4.2. Prediction performance in terms of feature engineering methods

To further evaluate the usefulness of different feature engineering

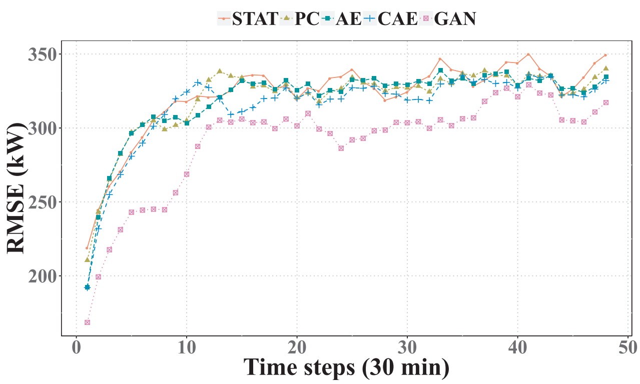

Fig. 16. The RMSEs along the 24 h prediction horizon using the multiple linear regression and direct multi-step prediction approach.

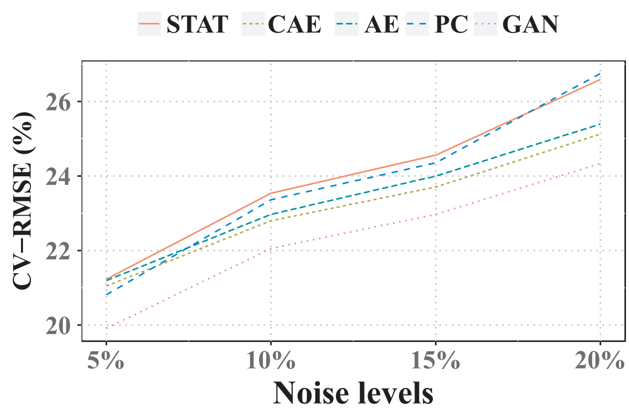

Fig. 17. Model performance at different masking noise levels.

methods for short-term building cooling load predictions, the RMSEs along the 24 h prediction horizon were calculated and visualized in Fig. 16 using the multiple linear regression and the direct multi-step ahead prediction approach [51]. The results are in accordance with those obtained in one-step ahead predictions. To summarize, given a supervised learning technique, the principal component-based and summarizing statistic-based features lead to the worst prediction accuracy. It is in accordance with expectations, as these two feature engineering methods can only derive linear features, i.e., principal components and summarizing mean values are in essence linear combinations of the original data. Such features cannot preserve high-level characteristics in the original data, e.g., the temporal dependency in time series data. By contrast, the features extracted by the other three deep learning-based methods are more useful for building energy predictions. The features extracted by feed-forward autoencoders can be regarded as nonlinear transformations of the original data. The resulting predictions are more accurate than their linear counterparts, i.e., features extracted by principal component analysis. As indicated by the research results, the features extracted by one-dimensional CAE can slightly enhance the prediction performance when using nonlinear modeling techniques. This may due to the unique capability of one-dimensional CAE in capturing temporal dependencies in time series data.

The features extracted by the GAN-based method typically lead to the best prediction performance. Such features are generated using a unique generative approach, i.e., rather than relying on the limited amount of training data alone, synthetic data are generated to facilitate the feature extraction process. Consequently, the feature extraction model (i.e., the discriminator D) can have a more complicated architecture without worrying about the over-fitting problem. In addition, once a reliable GAN model is developed, the output of generator G can be used to produce realistic samples on building cooling load profiles (as shown in Fig. 10). Such profiles can be further used for other building management tasks, e.g., simulating normal behaviors in building operations and calculating benchmarks for daily energy consumptions. Despite its great potential in analyzing building operation data, it should be mentioned that GANs are rather difficult to train. Hyperparameter tuning is a must to ensure the success of GAN model development. Further in-depth studies are needed to provide guidelines and references for GAN model development in the building field.

4.3. Practical values of deep learning-based feature engineering methods

The practical values of deep learning-based feature engineering methods can be summarized into two aspects, i.e., dimensionality reduction and data denoising. The first is obvious, as a smaller number of features are extracted to preserve the information in the original data. In such a case, the total number of observations needed for reliable model development is reduced, making it more flexible and less computationally expensive for practical applications.

In addition, the quality of building operation data can be poor due to the existence of many missing values, outliers and noisy measurements. As a result, the performance of prediction models developed may not be reliable and robust. To illustrate the usefulness of deep learning-based feature engineering methods in handling noisy data, an experiment was designed as follows: masking noises with zeros were randomly added to the original data with four levels of probabilities of 5%, 10%, 15% and 20%. Different sets of features were extracted and used for prediction model development. The XGB was used as the supervised learning technique. The resulting CV-RMSEs are shown in Fig. 17. It is evident that there is an increasing trend of CV-RMSEs as the masking noise levels increase. The two linear feature engineering methods, i.e., STAT and PC-based methods, still result in the worst performance. Meanwhile, better resulting performance can be obtained using features extracted by deep learning-based methods. The GAN-based method is especially useful to ensure the performance given noisy data. This may due to the generative modeling nature of GAN models. In essence, some of the synthetic samples generated by the generator G can be regarded as noisy samples. Consequently, the discriminator has the potential to extract high-level features for data denoising. In practice, such feature engineering methods are helpful to ensure the prediction performance using relatively low-quality building operation data.

5. Conclusions

Accurate and reliable predictions on building energy consumptions are very helpful for the development of building energy conservation measures. In practice, it can be very difficult to select or construct features from the original data for building energy predictions, as building operation data are highly correlated and noisy. Conventional feature engineering methods are heavily dependent on engineering experience, which are neither efficient nor effective for generalization and automation purposes.

To tackle this challenge, this study exploits the power of the state-of-the-art deep learning techniques in automatically extracting valuable features for building energy predictions. Three deep learning-based feature engineering methods have been developed based on the use of fully-connected autoencoders, one-dimensional convolutional autoencoders and generative adversarial networks respectively. Their potentials in extracting valuable features for predictive modeling have been assessed and compared with two conventional data-driven feature engineering methods. The research results show that deep learning-based feature engineering can lead to evident improvement in building energy predictions. The features extracted by the fully-connected autoencoders are in essence nonlinear transformations of the original data. It generally leads to better performance than its linear counterparts, i.e., principal components and summarizing statistics. Slightly better prediction performance can be obtained using features extracted by one-dimensional convolutional autoencoders. It indicates that one-dimensional convolutional autoencoders are capable of extracting useful temporal relationships in time series data for predictive modeling. More importantly, this study validates the usefulness of generative adversarial networks (GANs) in constructing high-level features from building operation data. The GAN-based feature engineering method adopts a generative approach for feature extraction, which is of great significance to tackle the challenge of extracting high-level features from limited and poor-quality building operation data. It should be noted that GANs can be very difficult to train and usually suffer from problems of non-convergence, vanishing gradients and unbalanced learning between the generator and discriminator. In this study, a trial-and-error approach was adopted for constructing GANs. Future study will focus on providing useful insights and guidelines for developing GANs for building energy data. In addition, the effectiveness of predictive modeling methods developed will be tested and validated using data from a variety of buildings to ensure the generalization performance [56]. As final words, the deep learning-based feature engineering methods proposed in this study are purely data-driven, which helps to fully automate the building energy prediction process. The research results help to bridge the knowledge gap between deep learning and building operations. It can be used as prototypes for developing more advanced tools for building energy predictions.

Acknowledgements

The authors gratefully acknowledge the support of this research by the Natural Science Foundation of SZU (No. 2017061), the Natural Science Foundation of Guangdong Province, China (No. 2016A030313011), and the National Natural Science Foundation of China (No. 71772125).

References

[1] Ramesh T, Prakash R, Shukla KK. Life cycle energy analysis of buildings: An overview. Energy Build 2010;42:1592–600.
[2] Teke A, Tumur O. Assessing the energy efficiency improvement potentials of HVAC systems considering economic and environmental aspects at the hospitals. Renew Sustain Energy Rev 2014;33:224–35.
[3] Fan C, Xiao F, Zhao Y. A short-term building cooling load prediction method using deep learning algorithms. Appl Energy 2017;195:222–33.
[4] Wang ZY, Wang YR, Srinivasan RV. A novel ensemble learning approach to support building energy use prediction. Energy Build 2018;159:109–22.
[5] Geronazzo A, Brager G, Manu S. Making sense of building data: New analysis methods for understanding indoor climate. Build Environ 2018;128:260–71.
[6] Wang YJ, Jin XQ, Du ZM, Zhu X. Evaluation of operation performance of a multi-chiller system using a data-based chiller model. Energy Build 2018;172:1–9.
[7] Ben-Nakhi AE, Mahmoud MA. Cooling load prediction for buildings using general regression neural networks. Energy Convers Manage 2004;45:2127–41.
[8] Lu L, Cai WJ, Xie LH, Li SJ, Soh YC. HVAC system optimization-in-building section. Energy Build 2005;37:11–22.
[9] Klein K, Herkel S, Henning HM, Felsmann C. Load shifting using the heating and cooling system of an office building: Quantitative potential evaluation for different flexibility and storage options. Appl Energy 2017;203:917–37.
[10] Thangavelu SR, Myat A, Khambadkone A. Energy optimization methodology for multi-chiller plant in commercial buildings. Energy 2017;123:64–76.
[11] Fan C, Xiao F, Zhao Y, Wang JY. Analytical investigation of autoencoder-based methods for unsupervised anomaly detection in building energy data. Appl Energy 2018;211:1123–35.
[12] Gao DC, Wang SW, Shan K, Yan CC. A system-level fault detection and diagnosis method for low delta-T syndrome in the complex HVAC systems. Appl Energy 2016;164:1028–38.
[13] Wei YX, Zhang XX, Shi Y, Xia L, Pan S, Wu JS, et al. A review of data-driven approaches for prediction and classification of building energy consumption. Renew Sustain Energy Rev 2018;82:1027–47.
[14] Ding Y, Zhang Q, Yuan TH. Research on short-term and ultra-short-term cooling load prediction models for office buildings. Energy Build 2017;154:254–67.
[15] Wang ZY, Wang YR, Zeng RC, Srinivasan RS, Ahrentzen S. Random forest based hourly building energy prediction. Energy Build 2018;171:11–25.
[16] Fan C, Xiao F, Wang SW. Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques. Appl Energy 2014;127:1–10.
[17] Zheng ZL, Zhuang Z, Lian ZW, Yu Y. Study on building energy load prediction based on monitoring data. Procedia Eng 2017;205:716–23.
[18] Zhang CB, Zhao Y, Zhang XJ, Fan C, Li TT. An improved cooling load prediction method for building with estimation of prediction intervals. Procedia Eng 2017;205:2422–8.
[19] Dahl M, Brun A, Andresen GB. Using ensemble weather predictions in district heating operation and load forecasting. Appl Energy 2017;193:455–65.
[20] Lim HS, Kim G. Prediction model of cooling load considering time-lag for pre-emptive action in buildings. Energy Build 2017;151:53–65.
[21] Ding Y, Zhang Q, Yuan TH, Yang F. Effect of input variables on cooling load prediction accuracy of an office building. Appl Therm Eng 2018;128:225–34.
[22] Wang HD. Empirical model for evaluating power consumption of centrifugal chillers. Energy Build 2017;140:359–70.
[23] Liu ZH, Tan HW, Luo D, Yu GB, Li J, Li ZY. Optimal chiller sequencing control in an office building considering the variation of chiller maximum cooling capacity. Energy Build 2017;140:430–42.
[24] Li Q, Meng Q, Cai JJ, Yoshino H, Mochida A. Predicting hourly cooling load in the building: a comparison of support vector machine and different artificial neural networks. Energy Convers Manage 2009;50:90–6.
[25] Guo Y, Nazarian E, Ko J, Rajurkar K. Hourly cooling load forecasting using time-indexed ARX models with two-stage weighted least squares regression. Energ Convs Manage 2014;80:46–53.
[26] Grolinger K, L'Heureux A, Capretz MAM, Seewald L. Energy forecasting for event venues: big data and prediction accuracy. Energy Build 2016;112:222–33.
[27] Cui C, Wu T, Hu MQ, Weir JD, Li XW. Short-term building energy model recommendation system: a meta-learning approach. Appl Energy 2016;172:251–63.
[28] Zhang C, Cao LW, Romagnoli A. On the feature engineering of building energy data mining. Sustain Cities Soc 2018;39:08–18.
[29] Roth J, Rajagopal R. Benchmarking building energy efficiency using quantile regression. Energy 2018;152:866–76.
[30] Schmidhuber J. Deep learning in neural networks: an overview. Neural Netw 2015;61:85–117.
[31] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;321:436–44.
[32] Li CD, Ding ZX, Zhao DB, Yi JQ, Zhang GQ. Building energy consumption prediction: An extreme deep learning approach. Energies 2017;10:1525.
[33] Cangelosi R, Goriely A. Component retention in principal component analysis with application to cDNA microarray data. Biol Direct 2007;2:1–21.
[34] Jolliffe I. Principal Component Analysis. New York: Springer; 2002.
[35] Zhong GQ, Wang LN, Ling X, Dong JY. An overview on data representation learning: From traditional feature learning to recent deep learning. J Finance Data Sci 2016;2:265–78.
[36] Bengio Y, Courville A, Vincent P. Representation learning: A review and new perspectives. IEEE Trans Pattern Anal Mach Intell 2013;35:1798–823.
[37] Goodfellow I, Bengio Y, Courville A. Deep learning. 1st ed. Cambridge, London, England: MIT press; 2016.
[38] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44.
[39] Witten IH, Frank E, Hall M, Pal CJ. Data mining: Practical machine learning tools and techniques. 4th ed. Singapore: Elsevier; 2017.
[40] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial networks. NIPS 2014:2672–80.
[41] Creswell A, White T, Dumoulin V, Arulkumaran K, Sengupta B, Bharath AA. Generative adversarial networks: An overview. IEEE signal processing magazine special issue on deep learning for visual understanding. 2017. p. 1–14.
[42] Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X. Improved techniques for training GANs. NIPS 2016, Barcelona, Spain.
[43] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. ICLR 2016, San Juan, Puerto Rico.
[44] Odena A. Semi-supervised learning with generative adversarial networks. In Data Efficient Machine Learning Workshop, ICML 2016, New York, USA.
[45] Amasyali K, El-Gohary NM. A review of data-driven building energy consumption prediction studies. Renew Sustain Energy Rev 2018;81:1192–205.
[46] Chen TQ, Guestrin C. XGBoost: a scalable tree boosting system. KDD 2016, San Francisco, CA, USA, August 13–17; 2016.
[47] Taieb SB, Hyndman R. A gradient boosting approach to the Kaggle load forecasting competition. Int J Forecasting 2014;30:382–94.
[48] Allaire JJ, Chollet F. Keras. Version 2.1.6. The Comprehensive R Archive. Network 2018.
[49] Heaton J. Introduction to neural networks for. JAVA. 2nd ed. Heaton Research Inc; 2008.
[50] R Development Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing. Vienna, Austria, ISBN 3-900051-07-0; 2008. URL < http://www.R-project.org > .
[51] Fan C, Wang JY, Gang WJ, Li SH. Assessment of deep recurrent neural network-based strategies for short-term building energy predictions. Appl Energy 2019;236:700–10.
[52] Reddy TA, Maor I, Panjapornpon C. Calibrating detailed building energy simulation programs with measured data-Part II: Application to three case study office buildings (RP-1051). HVAC&R Res 2007;13:221–41.
[53] Ashrae G. Guideline 14–2002. Atlanta, GA: Measurement of Energy and Demand Savings, American Society; 2002.
[54] Fan C, Xiao F, Yan CC, Liu CL, Li ZD, Wang JY. A novel methodology to explain and evaluate data-driven building energy performance models based on interpretable machine learning. Appl Energy 2018;235:1551–60.
[55] Ribeiro MT, Singh S, Guestrin C. Model-agnostic interpretability of machine learning. ICML Workshop on Human Interpretability in Machine Learning, New York, USA; 2016. arXiv: 1606.05386.
[56] Miller C, Meggers F. The building data genome project: An open, public data set from non-residential building electrical meters. Energy Procedia 2017;122:439–44.