Authors

Wang, Ran

Lu, Shilei

Feng, Wei

Publication Date

2020-03-01

DOI

10.1016/j.apenergy.2020.114561

Peer reviewed

A novel improved model for building energy consumption prediction based on model integration

Ran Wang[a,b], Shilei Lu[a,b,*], Wei Feng[b]

[a] School of Environment Science and Engineering, Tianjin University, 92 Weijin Road, Tianjin 300072, China
[b] Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

A B S T R A C T

Building energy consumption prediction plays an irreplaceable role in energy planning, management, and conservation. Constantly improving the performance of prediction models is the key to ensuring the efficient operation of energy systems. Moreover, accuracy is no longer the only factor in revealing model performance, it is more important to evaluate the model from multiple perspectives, considering the characteristics of engineering applications. Based on the idea of model integration, this paper proposes a novel improved integration model (stacking model) that can be used to forecast building energy consumption. The stacking model combines advantages of various base prediction algorithms and forms them into "meta-features" to ensure that the final model can observe datasets from different spatial and structural angles. Two cases are used to demonstrate practical engineering applications of the stacking model. A comparative analysis is performed to evaluate the prediction performance of the stacking model in contrast with existing well-known prediction models including Random Forest, Gradient Boosted Decision Tree, Extreme Gradient Boosting, Support Vector Machine, and K-Nearest Neighbor. The results indicate that the stacking method achieves better performance than other models, regarding accuracy (improvement of 9.5%–31.6% for Case A and 16.2%–49.4% for Case B), generalization (improvement of 6.7%–29.5% for Case A and 7.1%-34.6% for Case B), and robustness (improvement of 1.5%–34.1% for Case A and 1.8%–19.3% for Case B). The proposed model enriches the diversity of algorithm libraries of empirical models.

1. Introduction

The building and construction sector together are responsible for 36% of global final energy consumption and nearly 40% of total direct and indirect carbon dioxide emissions [1]. Building energy savings can be achieved by improving the building's dynamic energy performance in terms of sustainable construction management in urban-based built environments [2]. Meanwhile, building operations are information-intensive, due to the popularity of smart sensors and the adoption of intelligent building management systems [3]. A large amount of building operational data has been recorded to provide a basis for building performance analysis. Therefore, a promising strategy to address energy savings is to develop big data-driven approaches to building smart energy management.

Prediction models for building management systems have raised concerns. The predicted targets are mainly building energy consumption (building internal heat gains [4], building cooling loads [5], district heating load [6], electricity demand [7], peak power demand [8]),

* Corresponding author at: School of Environment Science and Engineering, Tianjin University, 92 Weijin Road, Tianjin 300072, China.
E-mail address: lvshilei@tju.edu.cn (S. Lu).

Nomenclature

Abbreviation

AI	Artificial intelligence
RMSE	Root mean squared error
MAE	Mean absolute error
MAPE	Mean absolute percentage error
CVRMSE	Coefficient of variation of the root mean square error
R^2	Coefficient of determination
x	Input
y	Output
P	Output of the train set of the base model
T	Output of the test set of the base model

Method

RF	Random forest
SVR	Support vector regression
kNN	k Nearest Neighbors
GBDT	Gradient boosting decision tree
XGBoost	Extreme gradient boosting
ANN	Artificial neural network
MLR	Multiple linear regression
ARIMA	Autoregressive integrated and moving average
BPNN	Back-propagation neutral network
GRNN	Generalized regression neural network
RBFNN	Radial basis function neural network
OLS	Ordinary least squares
MARS	Multivariate adaptive regression splines
GPR	Gaussian process regression
ES	Exponential Smoothing

indoor temperature (commercial buildings [9] and residential buildings [10]), thermal sensation votes [11], and system or unit performance indicators [12]. The service objectives of prediction models in the energy management system mainly include optimized control [13] and fault detection [14]. Optimized control includes matching the supply and demand of building energy [15], maintaining the indoor thermal comfort [9], and operating the unit and system in an optimal state (HVAC systems [16], energy recovery systems [17], and radiant floor systems [18]). With the support of sufficient training data, fault detection helps to distinguish whether the patterns of monitoring data are similar to those of the normal training data [19]. Hence, improving the accuracy, robustness, and generalization performance of the prediction algorithm is key to ensure efficient building operations [20].

The data-driven model has become the most popular method used in the field of building energy due to its low time consumption and good prediction performance [21]. Commonly used data-driven models can be classified as single prediction models, integration prediction models, and improved prediction models.

The single prediction model, which is the type of most traditional models, has single algorithm architectures such as support vector regression (SVR) [22], artificial neural network (ANN) [23], and multiple linear regression models (MLR) [24].

In contrast, integration prediction models integrate single prediction algorithms into a more accurate model by combining strategies, which can be divided into various types in view of the order between base models (in parallel or in series) and whether base models are the same kind of algorithms (homogeneous or heterogeneous integration) [25]. For example, the Random Forest (RF) model is a parallel homogeneous integration model, while Gradient Boosted Decision Tree (GBDT) and Extreme Gradient Boosting (XGBoost) models are series homogeneous integration models.

Improved prediction models use auxiliary algorithms or frameworks to make up for the deficiencies of the original prediction algorithm. There are generally three forms: (1) the pre-assisted algorithm, which improves the data quality to make up for the specific requirements of the prediction algorithm [26]; (2) the assisted optimization algorithm, which is used to perform hyperparameter tuning of a prediction algorithm [27]; and (3) nesting of an auxiliary improvement framework on the base prediction algorithm, to improve model performance [7]. Most of the existing research is based on single or integration models, which lack improvements in the essence of the algorithm. However, the improved prediction model can enrich the algorithm library and continuously improve the overall accuracy level of building energy forecasting, which is crucial for scheduling and managing energy usage [15]. Therefore, it is quite important to develop improved prediction models.

This paper proposes a novel prediction method for building energy prediction based on the principle of algorithm integration. The method is applied to two actual cases as a demonstration, and the results verify its reliability in short-term building energy prediction. Moreover, the method is compared with some state-of-the-art or popular prediction methods from the perspectives of robustness, accuracy, and generalization performance, and its superiority is verified. The proposed method enriches the library of energy consumption prediction models. Hence, this study has a unique significance.

2. Literature review

The most important two modules for the prediction model are model inputs and prediction algorithms. The model inputs section mainly summarizes the input types found in existing research and looks forward to the development trend. The prediction algorithm is summarized from three algorithm types: single prediction, integration prediction, and improved prediction. The literature related to the algorithms' performance comparison and performance evaluation dimensions is also reviewed.

2.1. Selection of model inputs

Input data can be classified into meteorological data, occupancy data, historical data, and time type information. Meteorological data are relatively easy to obtain by means of weather stations. Occupancy data mainly affects the building energy consumption by changing the energy supply status, which refers to human behaviors and building usage schedules. Due to the great uncertainty and high requirements for the monitoring system, few studies have directly taken occupancy data as input data for the prediction model; according to the literature, the proportion of meteorological and occupancy data used in related research articles is 60% and 29%, respectively [25]. In addition, some researchers utilized time type information (e.g., time of the day, day of the week, day type) to remedy the information lost from the omission of occupancy data [28]. Historical data such as historical energy is a popular input because it indicates the trend of the load profile in a mathematical way. The model inputs in most studies generally involve a combination of two or more of these types.

For example, meteorological data and historical energy consumption were used for daily building electricity forecasting [29]. Only historical energy consumption was used in yearly building electricity forecasting [30] and daily building electricity forecasting [31]. Meteorological data and day type were used in hourly electricity forecasting [32] and sub-hourly electricity forecasting [33]. Meteorological data and occupancy schedules were used in the prediction of heating/cooling consumption for a solar house [34]. Few studies used other variables as inputs; for example, indoor environmental factors (temperature and relative humidity) and meteorological data were used together to predict building cooling energy consumption [15]. Other than meteorological data, occupancy and hour-type/day-type pretreated air unit operation schedule were used for cooling load prediction [35].

Most researchers collected input data based on their knowledge of the prediction model and data availability [25]. Whether it is in the establishment of the model or the actual maintenance of the model later, collection of meteorological data is relatively easy. Additionally, day types and historical data are deterministic prior information; so the use of these three types of data is still dominant. Meanwhile, the number of inputs for each input data type has increased as modeling techniques have developed [5]. For example, meteorological data were only related to outdoor temperature in early studies [36], but this gradually expanded to outdoor temperature, relative humidity, and solar radiation [37]. Currently, wind speed and wind direction also serve as indicators [15].

2.2. Evolving prediction models

2.2.1. Single prediction models

The single prediction model's main feature is that it consists of only the base prediction algorithm. Early research related to building energy predictions generally involves single prediction models, which are mainly divided into statistical and artificial intelligence models [38]. Statistical models are an established, simple tool for long-term prediction [39]. MLR, Exponential Smoothing (ES), and auto-regressive integrated and moving average (ARIMA) models are popular statistical approaches that are applied in the prediction model. The MLR model was used to forecast the daily peak load [40] and monthly electricity demand [41]. The ES model was used for hourly load forecasting, with a lead time of 1–24 h [42]. The ARIMA model was used to predict hourly electricity load and daily peak load [43]. The kNN is also a simple and effective statistical model, widely applied to wind speed forecasting [44], electricity forecasting [45], and solar power forecasting [46].

Due to high accuracy, artificial intelligence (AI) algorithms are widely used; the transition from statistical to AI methods occurred around 1991 to 2001 [39]. The SVR and the ANN series of algorithms are commonly used predictive models [47]. SVR is based on the structural risk minimization principle, which performs well in time series and non-linear prediction [48]. It was first used in the field of building energy consumption forecasting in 2005 [49]. ANN is a non-linear statistical learning technique inspired by biological neural networks, applied to various types of building energy consumption forecasts, such as overall building energy consumption, cooling and heating loads, and electricity consumption. Back-propagation neural network (BPNN) and Generalized regression neural network (GRNN) are two representative types of ANN. Ben et al. used GRNN to predict the cooling load [36]. Ekici et al. proved the reliability and accuracy of BPNN in building heating load forecast [50].

2.2.2. Integration prediction models

A more advanced data-driven method called integration learning was introduced in the early 1990s. Integration learning is also called fusion learning, aggregation, combination, ensemble, and other names [25]. An integration model is defined as a framework that combines the advantages of multiple single models to improve overall performance. It is divided into two types: heterogeneous and homogeneous. Heterogeneous models use the same dataset to construct single models by training different algorithms or the same algorithm with different parameter settings [51]. Chou et al. constructed an integration model based on SVR and ANN [52]. Chae et al. constructed an integration model based on three ANN algorithms and applied it to predict sub-hourly electricity usage in commercial buildings [28]. In contrast, homogeneous models are constructed by the same single models on different training sets. RF has been effectively applied to peak power demand [8], electricity load forecasting [53], and heating and cooling loads [54]. In recent years, GBDT and XGBoost have also gradually been applied in the field of building energy. The GBDT model exhibits the highest performance in the prediction of energy consumption by appliances in a low-energy house [55], electricity load forecasting for utility energy management systems [56], and electricity load forecasting for utility energy management systems [57]. The XGBoost model has been used to construct a prediction model for early detection of faults in HVAC systems [58] and building energy performance grading [59]. In general, integrated algorithms are becoming increasingly popular in the field of energy prediction.

2.2.3. Improved prediction models

Some studies have focused on the improved prediction model to achieve better accuracy.

Auxiliary algorithms can be applied to improve data quality before the prediction algorithm is established. For example, Ding et al. used K-means and hierarchical clustering methods to classify input variables to improve prediction accuracy [60]. Yuan et al. proposed a sample data selection method based on a grey correlation method integrated with an entropy weight method; the result demonstrated that the accuracy of BPNN had improved [26]. Ding et al. divided the sample data by ten-fold cross-validation to improve the accuracy of the SVR model in short-term and ultra-short-term predictions of cooling load [61]. A hybrid SVR was applied to predict the hourly electric demand intensity; the multi-resolution wavelet decomposition was introduced to divide the initial series into several parts, to alleviate the interferential influence on modeling [62].

Optimization algorithms can be used for hyperparameter tuning to improve base model performance. For example, Li et al. applied an improved particle swarm optimization algorithm to adjust the structure weights and threshold values of ANN [63]. Zhong et al. proposed a novel vector field-based SVR method, which improves the performance of SVR, including accuracy, robustness, and generalization capabilities through multi-distortions in the sample data space or high-dimensional feature space mapped by a vector field to find the optimal feature space [15]. An evolutionary-based ANN algorithm has been proposed for short-term load forecast of electricity, and optimal network parameters are found to reduce the forecasting error [27].

Some integration strategies and frameworks can be used to improve the structure of the base algorithm to enhance its performance. For example, Fan et al. exploited the potential of deep learning and compared its performance in cooling load prediction with typical feature extraction methods and popular prediction techniques in the building field. Results showed that deep learning can enhance the predictive performance, especially when used in an unsupervised manner [64]. Jatin et al. proposed a Long Short Term Memory based deep learning framework to forecast electricity demand by taking care of long-term historical dependencies, and proved the method's effectiveness by comparing it with ANN and SVR [7]. Alessandro et al. proposed a Bayesian deep learning-based method to predict electricity price, and proved the method's robustness in out-of-sample conditions [65]. On the whole, improved prediction models enrich the algorithm library of the prediction model and promote the accuracy of the energy prediction field overall.

2.3. Comparison of prediction models

From the perspective of algorithm types, existing studies compare the same, as well as different, types of models.

Many studies have conducted comparisons between the same type of model. For example, Li et al. compared SVR with several ANN models for predicting hourly building cooling load [66]. Massana et al. used SVR, MLR, and ANN to predict short-term load for non-residential buildings [67]. Wang et al. compared SVR and three ANN models (BPNN, radial basis function neural network [RBFNN], and GRNN) for predicting hourly residential electricity use [68]. These studies have shown that SVR improved building energy use prediction better than other AI-based prediction methods. ANN was compared with regression models for annual urban residential buildings' energy consumption [69] and HVAC hot water energy consumption [70]; both studies indicated that ANN could perform better than regression methods for short-term forecasting. Comparisons have also been made between single and integrated models. For example, three neural network models and their integrated forms were used for heating consumption prediction, and the results showed that the integrated model has better prediction accuracy [71]. Ahmad et al. compared ANN and RF models for predicting the hourly HVAC energy consumption of a hotel, and ANN performed with marginally better precision than RF, but RF has an advantage in processing multi-dimensional complex data due to its ease of tuning and modeling [72]. RF was compared with RT and SVR to validate its superiority in building energy prediction [53]. One example of a comparison between a single model, integrated model, and hybrid model is the study by Zhong et al., which compared an improved SVR algorithm with the MLR, BPNN, SVR, deep learning, and GBDT models [15].

It is important to note that most prior research studies have only compared model performance in terms of precision, although a small number of research studies have compared model performance from multiple perspectives. For example, SVR and three ANN models (including BPNN, RBFNN, and GRNN) were applied to hourly cooling load forecasting of an office building; the result demonstrated that the SVR and GRNN could achieve better accuracy and generalization than the BPNN and RBFNN [66]. Wang et al. compared five models (SVR, ANN, RF, GBDT, and XGBoost) with respect to interpretability, accuracy, robustness, and computational efficiency when applied to hourly heating energy consumption. BPNN exhibited the lowest precision, efficiency, robustness, and interpretability, while RF showed the highest overall performance, with the highest accuracy, robustness, and interpretability. However, RF efficiency was less than the XGBoost model [73]. Ostergard et al. compared six prediction models with respect to accuracy, efficiency, ease-of-use, robustness, and interpretability. These models included linear regression with ordinary least squares (OLS), RF, SVR, multivariate adaptive regression splines (MARS), Gaussian process regression (GPR), and ANN. The comparison shows that GPR produced the best precision and robustness, and was easy to implement, but it became inefficient for large training sets compared to ANN and MARS [74]. Both studies have shown there was no consensus on a "best" model after considering all their performances. Zhong et al. verified the performance of an improved SVR algorithm from three aspects of prediction accuracy, generalization ability, and robustness by comparing it with commonly used data-driven models and state-of-the-art models [15]. Cai et al. compared two deep neural network models with the Seasonal ARIMA model for accuracy, computational efficiency, generalizability, and robustness [20]. Fan et al. investigated and compared the usefulness of advanced recurrent neural network-based strategies for building energy predictions in terms of prediction accuracy and computation load [75].

2.4. A summary of the previous research

Most studies have established prediction models using base data mining algorithms. Among them, the traditional single prediction model is the most widely used because of the simple algorithm involved. The integrated model is becoming increasingly popular for building energy use prediction due to its remarkably improved prediction accuracy [25]. The integrated model uses multiple base models to predict the results, and the diversity among these base models will reduce the prediction error of the overall system. In terms of building energy consumption prediction, the accuracy improvement of the integrated model in the reviewed research can be up to 50% based on the MAPE index [8] and up to 4.9% based on the RMSE index [71].

Improved algorithms can effectively improve the accuracy of prediction models; however, related research is rarely compared with basic data-mining algorithms. Therefore, it is of great significance to propose an improved prediction model with better performance for enriching the empirical model library of energy consumption prediction.

The horizontal comparison between different models can be conducive to the intuitive display of relative advantages and disadvantages, and most of the current research has evaluated models based solely on accuracy. Moreover, from the perspective of practical engineering applications, the model performance should be evaluated from multiple perspectives, including accuracy, generalization, and robustness, in

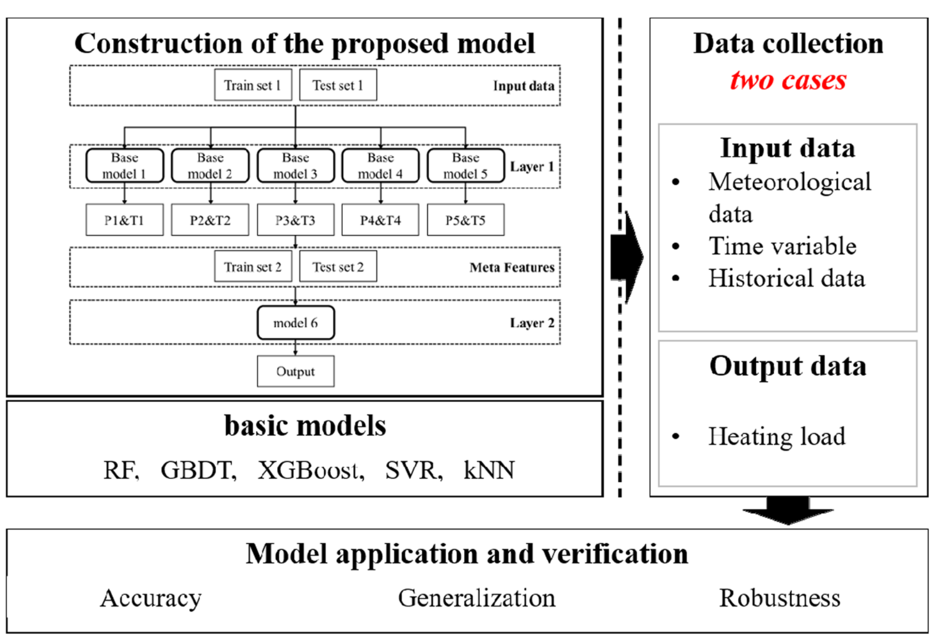


Fig. 1. The research framework.

order to adapt to the status of prediction models becoming more and more important in energy management systems.

3. Framework and methodologies

This study seeks to contribute to the existing state of the art by focusing on the following:

- This paper proposes a novel improved prediction algorithm. Its core is to build an integration framework and apply it to the basic models to improve overall prediction performance. The improved prediction model includes two main features: it enables comprehensive data observation and reduces overfitting. Based on the characteristics of different prediction algorithms, it can observe sample data from different spatial and structural perspectives. This study combines the observations of various base models into the form of "meta-features" to enable the overall model to more comprehensively observe the sample data. And this method reduces overfitting by distorting the sample data space.
- The superiority of the proposed model is verified by several state-of-the-art or popular prediction models. Model performance is evaluated from three dimensions: accuracy, generalization, and robustness.

3.1. Research framework

This study is dedicated to obtaining a novel method of high precision, high robustness, and high generalization capabilities for building energy prediction. Subsequently, the proposed method is applied to a real case to establish a prediction model and verify its effectiveness. This section focuses on the construction of the novel model and case introduction. The research framework is shown in Fig. 1.

3.2. Proposed method

3.2.1. Algorithm framework

Based on the idea of model fusion, this paper presents a new energy consumption prediction model. Single models can observe data in different spatial and structures angles, the proposed model (called the stacking model hereafter) can synthesize the observations of all single models to achieve an improved prediction performance by constructing a novel integration framework. Overall, the stacking model has a structure with two layers, as shown in Fig. 2. The first layer is the construction of base models, and their output constitutes "meta-features" as an input to the second layer. The second layer is a combined strategy model.

The detailed components are as follows:

Each base model trains the training set to predict the tag columns of train and test, and the predicted results of the training set and test set are taken as P and T, respectively. For base model 1 to base model 5, corresponding P and T are obtained, respectively.

$$\begin{pmatrix} \vdots \\ \vdots \\ P_1 \\ \vdots \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vdots \\ T_1 \\ \vdots \\ \vdots \end{pmatrix} \tag{1}$$

P_1–P_5 and T_1–T_5 are combined separately to obtain a new training set "trainset 2" and test set "testset 2".

$$\begin{pmatrix} \vdots \\ \vdots \\ P_1 \\ \vdots \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vdots \\ P_2 \\ \vdots \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vdots \\ P_3 \\ \vdots \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vdots \\ P_4 \\ \vdots \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vdots \\ P_5 \\ \vdots \\ \vdots \end{pmatrix} \Rightarrow \overbrace{\begin{pmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ P_1 & P_2 & P_3 & P_4 & P_5 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}}^{Trainset2}$$

$$\begin{pmatrix} \vdots \\ \vdots \\ T_1 \\ \vdots \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vdots \\ T_2 \\ \vdots \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vdots \\ T_3 \\ \vdots \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vdots \\ T_4 \\ \vdots \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vdots \\ T_5 \\ \vdots \\ \vdots \end{pmatrix} \Rightarrow \overbrace{\begin{pmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ T_1 & T_2 & T_3 & T_4 & T_5 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}}^{Testset2} \tag{2}$$

The model of the second layer "model 6" is used to train "Trainset2" and predict "Testset2" to get the final output.

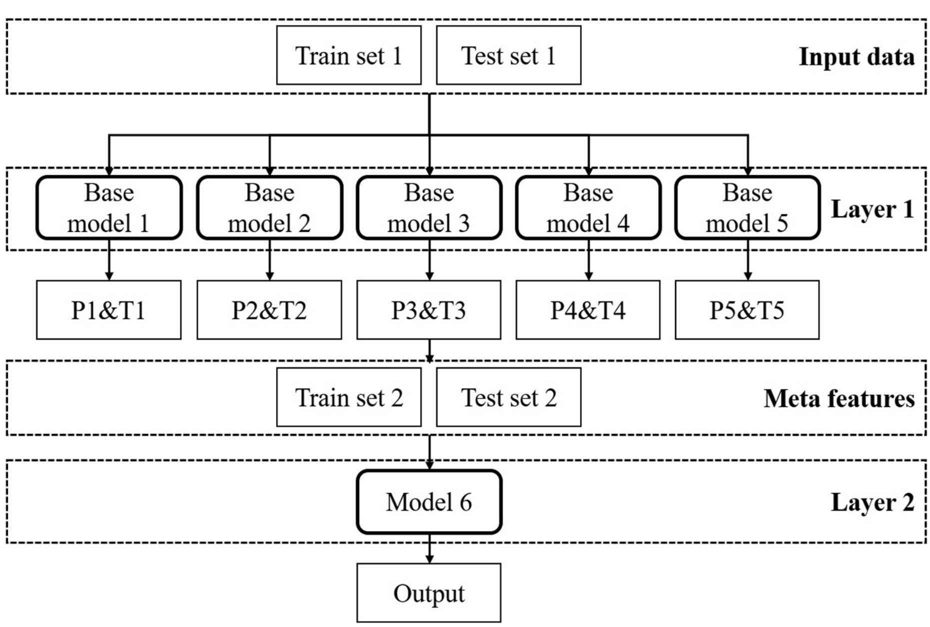


Fig. 2. The global working flow of the stacking method.

$$\overbrace{\begin{pmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ P_1 & P_2 & P_3 & P_4 & P_5 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}}^{Trainset2} \xRightarrow{\text{train}} \overbrace{\begin{pmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ T_1 & T_2 & T_3 & T_4 & T_5 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}}^{Testset2} \xRightarrow{\text{predict}} \overbrace{\begin{pmatrix} \vdots \\ \vdots \\ \text{pred} \\ \vdots \\ \vdots \end{pmatrix}}^{Output} \tag{3}$$

3.2.2. Construction of base models

In order to avoid overfitting, fivefold cross-validation is used for each base model. As shown in Fig. 3, divide “trainingset 1” into five non-intersecting sets and mark them as $train_a$ to $train_e$. Assume that the training set is a matrix of M rows and N columns. The matrix of the five train sets is as shown in Eq. (4).

$$\overbrace{\begin{pmatrix} x_{11} & \cdots & x_{1N} \\ \vdots & \ddots & \vdots \\ x_{1/5M1} & \cdots & x_{1/5MN} \end{pmatrix}}^{train_a} \overbrace{\begin{pmatrix} x_{1/5M1} & \cdots & x_{1/5MN} \\ \vdots & \ddots & \vdots \\ x_{2/5M1} & \cdots & x_{2/5MN} \end{pmatrix}}^{train_b}$$
$$\overbrace{\begin{pmatrix} x_{2/5M1} & \cdots & x_{2/5MN} \\ \vdots & \ddots & \vdots \\ x_{3/5M1} & \cdots & x_{3/5MN} \end{pmatrix}}^{train_c} \overbrace{\begin{pmatrix} x_{3/5M1} & \cdots & x_{3/5MN} \\ \vdots & \ddots & \vdots \\ x_{4/5M1} & \cdots & x_{4/5MN} \end{pmatrix}}^{train_d}$$
$$\overbrace{\begin{pmatrix} x_{4/5M1} & \cdots & x_{4/5MN} \\ \vdots & \ddots & \vdots \\ x_{M1} & \cdots & x_{MN} \end{pmatrix}}^{train_e} \tag{4}$$

Taking base model 1 as an example, base model 1 is trained by the combination of $train_a$–$train_d$ to build Model_e. And the dataset $train_e$ is predicted by Model_e to obtain $Pred_e$ as shown in Eq. (5).

$$\overbrace{\begin{pmatrix} x_{11} & \cdots & x_{1N} \\ \vdots & \ddots & \vdots \\ x_{1/5M1} & \cdots & x_{1/5MN} \end{pmatrix}}^{train_a} \cdots \overbrace{\begin{pmatrix} x_{3/5M1} & \cdots & x_{3/5MN} \\ \vdots & \ddots & \vdots \\ x_{4/5M1} & \cdots & x_{4/5MN} \end{pmatrix}}^{train_d} \xRightarrow{train} \overbrace{\begin{pmatrix} x_{4/5M1} & \cdots & x_{4/5MN} \\ \vdots & \ddots & \vdots \\ x_{M1} & \cdots & x_{MN} \end{pmatrix}}^{train_e} \xRightarrow{predict} \overbrace{\begin{pmatrix} pred_{4/5M} \\ \vdots \\ pred_M \end{pmatrix}}^{Pred_e} \tag{5}$$

Similarly, Model_a–Model_d are built, and $Pred_a$–$Pred_d$ are obtained by using $train_a$, $train_b$, $train_c$, and $train_d$ as prediction sets, respectively.

$$\overbrace{\begin{pmatrix} pred_1 \\ \vdots \\ pred_{1/5M} \end{pmatrix}}^{Pred_a} \overbrace{\begin{pmatrix} pred_{1/5M} \\ \vdots \\ pred_{2/5M} \end{pmatrix}}^{Pred_b} \overbrace{\begin{pmatrix} pred_{2/5M} \\ \vdots \\ pred_{3/5M} \end{pmatrix}}^{Pred_c} \overbrace{\begin{pmatrix} pred_{3/5M} \\ \vdots \\ pred_{4/5M} \end{pmatrix}}^{Pred_d} \tag{6}$$

Finally, $Pred_a$–$Pred_e$ is combined to form P_1.

$$\overrightarrow{Pred_a} + \overrightarrow{Pred_b} + \overrightarrow{Pred_c} + \overrightarrow{Pred_d} + \overrightarrow{Pred_e} = \begin{pmatrix} Pred_a \\ Pred_b \\ Pred_c \\ Pred_d \\ Pred_e \end{pmatrix} = \begin{pmatrix} \vdots \\ \vdots \\ P_1 \\ \vdots \\ \vdots \end{pmatrix} \tag{7}$$

Using the established Model_a–Model_e to predict the testset to get $Test_a$–$Test_e$ separately, and finally average the five results to get T_1.

$$\overbrace{\begin{pmatrix} Test_{1a} \\ \vdots \\ Test_{Pa} \end{pmatrix}}^{Test_a} \overbrace{\begin{pmatrix} Test_{1b} \\ \vdots \\ Test_{Pb} \end{pmatrix}}^{Test_b} \overbrace{\begin{pmatrix} Test_{1c} \\ \vdots \\ Test_{Pc} \end{pmatrix}}^{Test_c} \overbrace{\begin{pmatrix} Test_{1d} \\ \vdots \\ Test_{Pd} \end{pmatrix}}^{Test_d} \overbrace{\begin{pmatrix} Test_{1e} \\ \vdots \\ Test_{Pe} \end{pmatrix}}^{Test_e} \tag{8}$$

$$(\overrightarrow{Test_a} + \overrightarrow{Test_b} + \overrightarrow{Test_c} + \overrightarrow{Test_d} + \overrightarrow{Test_e})/5 = \begin{pmatrix} \vdots \\ \vdots \\ P_1 \\ \vdots \\ \vdots \end{pmatrix} \tag{9}$$

3.2.3. The selection of base models

These base models are selected based on two principles: popularity and diversity. First, all the selected algorithms should have been widely used in solving complex modeling and prediction problems, and studies should have proven their performance to be encouraging. Second, the maximized integration diversity will make the integration results more robust and more accurate [8], and it is best to “cross the space” between the selected base models. That means algorithms with large differences in principle should be selected. Therefore, this study selects five commonly used models (RF, kNN, SVR, GBDT, and XGBoost) as base models. These differ in their non-linearity handling abilities, model architectures, and inference mechanisms.

The RF model was developed by Breiman in 2001 for both

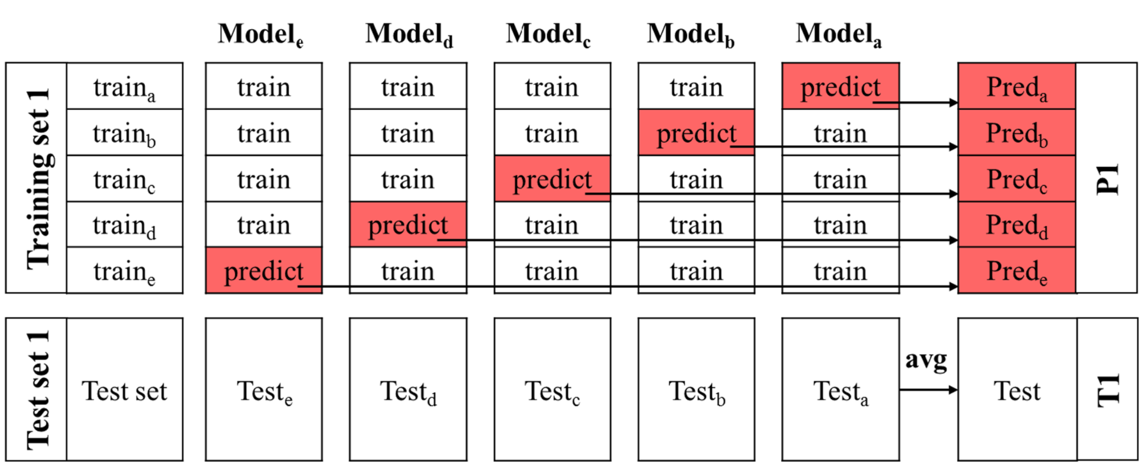


Fig. 3. The construction of the base model (Taking Base model 1 as an example).

classification and regression problems [76]. It consists of many decision regression trees but is not a simple average of the predictions of all decision trees. Its four features are bootstrap resampling, random feature selection, out-of-bag error estimation, and full-depth decision tree growing. Repeated sampling of the original datasets generates each of the regression trees. The samples of about one-third are not extracted at each repeated sampling, which forms a control dataset. RF is not easy to overfit and has good noise immunity [77].

The kNN model is a non-parametric learning algorithm used for either classification or regression. kNN regression uses the averaging method, which is the average output of the most recent K samples, as the regression prediction. It is non-parametric, as it does not learn an explicit mapping relationship between inputs and outputs. The parameter k, which defines the number of considered neighboring observations, is important for model performance. The larger the k value, the better the generalization ability of the model, but it is easy to fit; the smaller the k value, the better the fitting effect of the model, but the generalization ability is not enough. kNN is regarded as one of the simplest learning algorithms [78].

The concept of the SVR model derives from the computation of a linear regression function in a high-dimensional feature space, where the input data are mapped through a non-linear function [79]. The most prominent advantage of SVR is the uniqueness and global optimality of the generated solution, as it does not require non-linear optimization with the risk of sucking in a local minimum limit. The Gaussian radial basis function is adopted as the kernel function because it can map the low-dimensional input space into the high-dimensional space, and only one parameter needs to be set [37].

GBDT is also called multiple additive regression trees (MART) or gradient boosting machine (GBM), and it is an iterative decision tree algorithm [80]. Its implementation logic is to build the weak learner (regression tree) in turn and try to reduce the deviation of the combiner. GBDT, based on numerical optimization, uses the fastest descent method to solve the optimal solution of loss function; fitting the negative gradient using the regression tree and calculating the step length using the Newton method. This makes it possible to reduce the loss function as fast as possible for each training and converge to the local optimal or global optimal solution as quickly as possible [81].

XGBoost is an improved algorithm based on the GBDT algorithm. It has the following improvements: First, XGBoost is based on analytic thinking. The loss function is expanded to the second-order derivatives to obtain the analytic solution as the gain to establish trees so that the loss function is optimal. Second, XGBoost adds regular terms to control model complexity. From the point of bias-variance trade-off, the regular terms reduce the model's variance and prevent overfitting. Third, XGBoost supports parallel learning and is relatively faster in computing. The detailed calculation formula is shown in [82].

3.3. Model performance evaluation

3.3.1. Accuracy

To ensure the reliability of the evaluation results, a variety of accuracy metrics can be used, including mean absolute error (MAE), root mean squared error (RMSE), mean absolute percentage error (MAPE), and the coefficient of variation of root mean square error (CVRMSE). Each of these indicators has a different emphasis. MAE is based on absolute error, which can visually show the average distance between the predicted value and the actual value. RMSE is used to identify large errors and evaluate the fluctuation of model response regarding variance. The metric punishes large errors severely because it geometrically amplifies the error [53]. MAPE expresses accuracy in percentage and reduces the effect of absolute errors caused by individual outliers. CVRMSE normalizes the prediction error and can provide a unitless metric that is more convenient to compare [38]. For all four indicators, the smaller the value, the better the model performance. In this article, four indicators are used in conjunction with each other. MAE is mainly used to show the difference between absolute errors, RMSE is used to identify large errors, and MAPE and CVRMSE are mainly used to compare the accuracy differences between different models.

$$CVRMSE = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}}{\bar{y}} \times 100\% \tag{10}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{11}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \tag{12}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{13}$$

where y_i, $\hat{y}_i$, and $\bar{y}$ represent the actual value, the predicted value, and the average actual value, respectively.

3.3.2. Generalization

Generalization performance refers to the model's ability to predict samples beyond the training range.

(a) (b)

Fig. 4. Exterior view of the case buildings: (a) Case A; (b) Case B.

3.3.3. Robustness

In the field of machine learning, robustness is defined as the ability of the model to resist external environmental disturbances to ensure the stability of its working performance. For building energy, there are many reasons why the collected operational data may deviate from reality, such as monitoring system failure, instrument damage, sudden stopping of the unit, and so on. Additionally, in the actual operation of the prediction model, weather forecasting is generally used to obtain meteorological data, but the error in weather forecasting will affect the model performance. Many studies have used the method of adding different intensities of noise to the testing data to measure the model's robustness. For example, Gaussian white noise was used to continuously enhance noise intensity [20]. R^2 reflects the proportion of the total variation of the dependent variable that can be explained by the independent variable. Therefore, it can be used as the main indicator of the model's robustness.

4. Case studies

4.1. Building information

Operation data retrieved from two educational buildings in the coastal city of Tianjin, China, is employed for the case study. The case study buildings mainly contain classrooms for students and offices for university staff. Case A is a three-star green building with three stories, and Case B is a conventional building with four stories. Their elevations are shown in Fig. 4. Table 1 summarizes the basic information of the case study buildings, including shape coefficient, insulation level of the envelope, and window-to-wall ratio. The insulation level of Case A is much better than that of Case B. Both buildings use district heating to maintain an indoor thermal environment during winter, and the heating energy is supplied by the energy station in the school.

4.2. Input data and data collection

The input variables can be divided into three general categories: (1) meteorological data (including outdoor dry bulb temperature, wet bulb temperature, relative humidity, wind direction, wind speed, air pressure, horizontal total radiation), (2) time variable (hour of the day, day type), and (3) historical data (energy consumption at the same time as the previous day). We conducted an on-site collection of the actual operational data, and obtained integrated datasets of Case A ranging from Dec. 1, 2017 to Jan. 20, 2018, with a time interval of 1 h, and Case B ranging from Dec. 5, 2017 to Dec. 14, 2017, with a time interval of 0.5 h. Fig. 5 shows the meteorological data and heating load data for the two buildings.

The total building heating load can be calculated based on the water flow rate, the supply, and the return water temperature. A portable flow meter was installed on the return manifold at the thermal inlet of each case building. The two probes of the wall-mounted dual-temperature self-recording instrument were placed inside the insulation layer of the water supply and return pipes. Fig. 6 shows the field test, and Table 2 shows the instrument parameters such as measurement accuracy.

Meteorological data were obtained from an on-campus weather station located approximately 100 m from the test building. The weather station consists of a complete set of weather sensors, including temperature, humidity, wind speed, wind direction, rainfall, and solar intensity. Table 2 shows their measurement accuracy.

4.3. Model implementation

The operational data in the previous section has revealed that there are differences in the energy use characteristics of the two case buildings; so we can apply the prediction algorithm to each building separately to verify the method scalability. For each case, the entire dataset can be divided into training and testing datasets, with proportions of 70% and 30%, respectively. The training set can be used to construct the prediction model, and the test set, which contains values outside the training set, can be used to examine the generalization ability. Prediction models were implemented in the Python environment (Ver. 3.6). The base model of the first layer is RF, XGBoost, GBDT, SVR, and kNN, and the combined model of the second layer is GBDT. The main parameters affecting the RF model include the maximum depth of the tree (MD), the number of trees (NT), and the maximum of features (MF). Compared to the RF model, the GBDT model includes the hyperparameter of the learning rate (LR) as well. When constructing a tree, XGBoost can subsample the original datasets. Unlike RF, the sample here is not placed back. The hyperparameters that affect the XGBoost model are mainly MD, NT, LR, and subsample. The SVR model has two crucial parameters: c and γ. The *c* is the penalty factor, which is the tolerance for the error. The γ is the coefficient of the kernel function, which implicitly determines the distribution of the data after mapping to the new feature space [73]. The *k* and the *p* of the Minkowski distance formula are the main hyperparameters affecting the kNN model [78]. To intuitively prove the improvement effect of the improved algorithm, the prediction results of base models also output separately. The base models in the stacking model were tuned in turn, and their optimized hyperparameters are summarized in Table 3.

5. Results and discussion

5.1. Model accuracy

Fig. 7 shows fitting results between the predicted and the corresponding measured value for each prediction model. The scatter of measured data—predicted data—is distributed on the sides of the baseline. Furthermore, the stacking model fits better than the other models, with R^2 equal to 0.86 and 0.92 for cases A and B, respectively. The R^2 of RF, GBDT, SVR, XGBoost, and kNN are 0.79, 0.82, 0.79, 0.82, and 0.84, respectively, for Case A, and 0.73, 0.85, 0.9, 0.89, 0.96, respectively, for Case B. Fig. 8 shows the relative error distribution of all models for the two cases. When based on the mean relative error, the stacking method achieves higher accuracy than all the other five models, with accuracy improvement being about 9.5%–31.6% for Case A and 16.2%–49.4% for Case B. The x-axis is the heating load, so it can visually show the variation of the relative error with the heating load level. As shown, the error of the stacking model is not the smallest compared to other models when the heating load was relatively low, but the error is smallest when the heating load is in the middle position. The pattern of actual heating loads is considered to be a normal distribution with the characteristics of "more in the middle and less at both ends" (as shown in Fig. 9). Therefore, from the perspective of engineering practice, the stacking model has better practicability in terms of cumulative error. Table 4 summarizes four error metrics of each model calculated using Eqs. (10)–(13). For the stacking model, the CVRMSE, RMSE, MAE, and MAPE are 10.60%, 23.53, 16.14 and 7.66%, respectively, for Case A, and 8.96%, 13.81, 10.61, and 7.51%,

Table 1
Basic information on the case study buildings.

Items		Case A	Case B
Size (m^2)		10,762.0	12,236.2
Insulation type		Self-insulation	External insulation
Shape Coefficient		0.18	0.22
Window-wall ratio	South	0.34	0.53
	East	0.13	0.29
	West	0.15	0.42
	North	0.43	0.37
U-value (W/(m^2·K))	Roof	0.17	0.55
	External wall	0.46	0.55
	External window	2.5	2.5

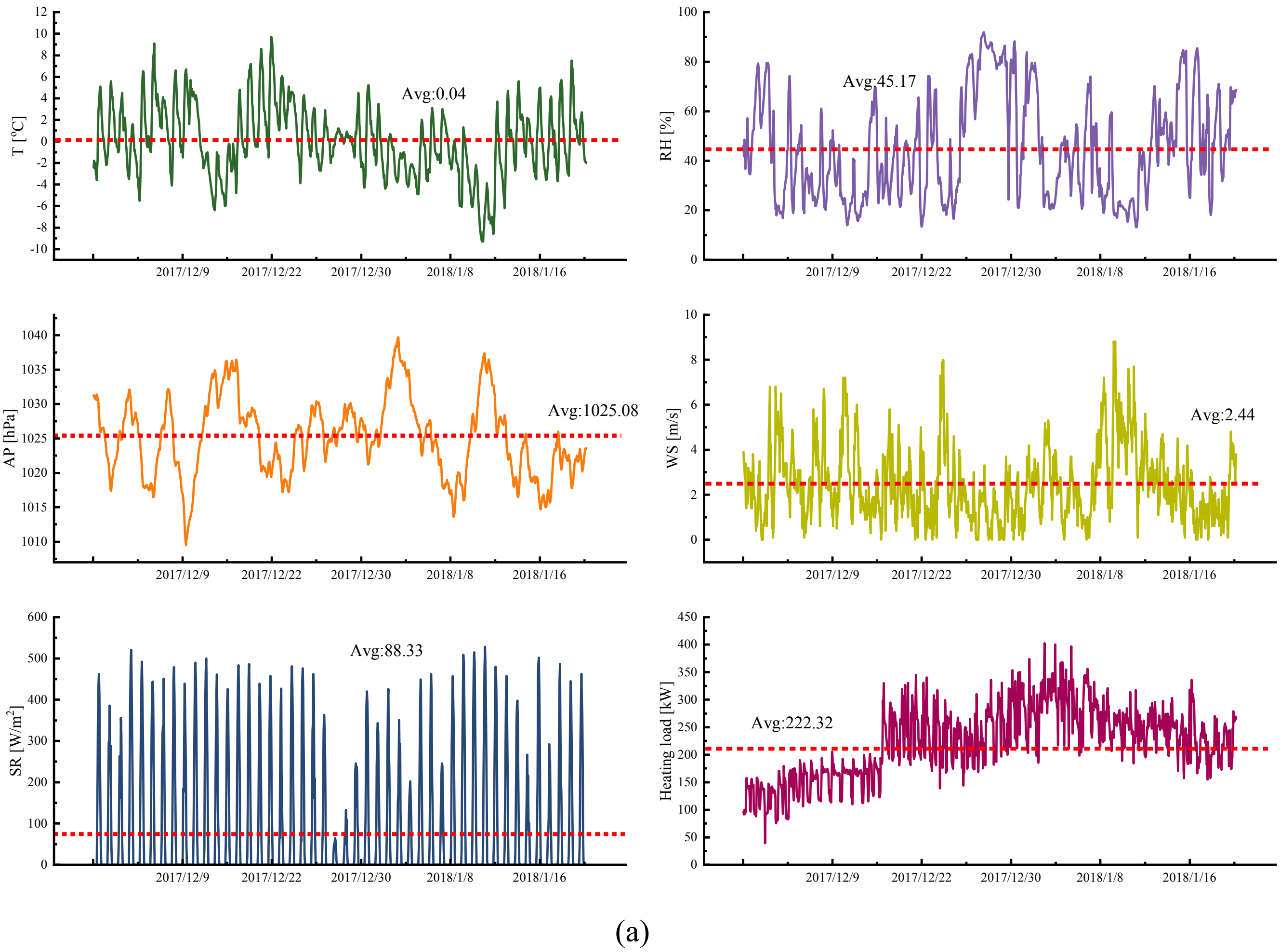


Fig. 5. The meteorological data and the energy load: (a) Case A; (b) Case B.

respectively, for Case B. Overall, the predictive performance for Case B is better than that of Case A. One potential reason is that the envelope insulation of Case A is better than that of Case B, so its heating load is relatively less affected by outdoor weather conditions. In any case, all four error metrics for both cases reveal that the proposed improved integrated algorithm has the best accuracy level compared to other advanced predictive models.

5.2. Model generalization

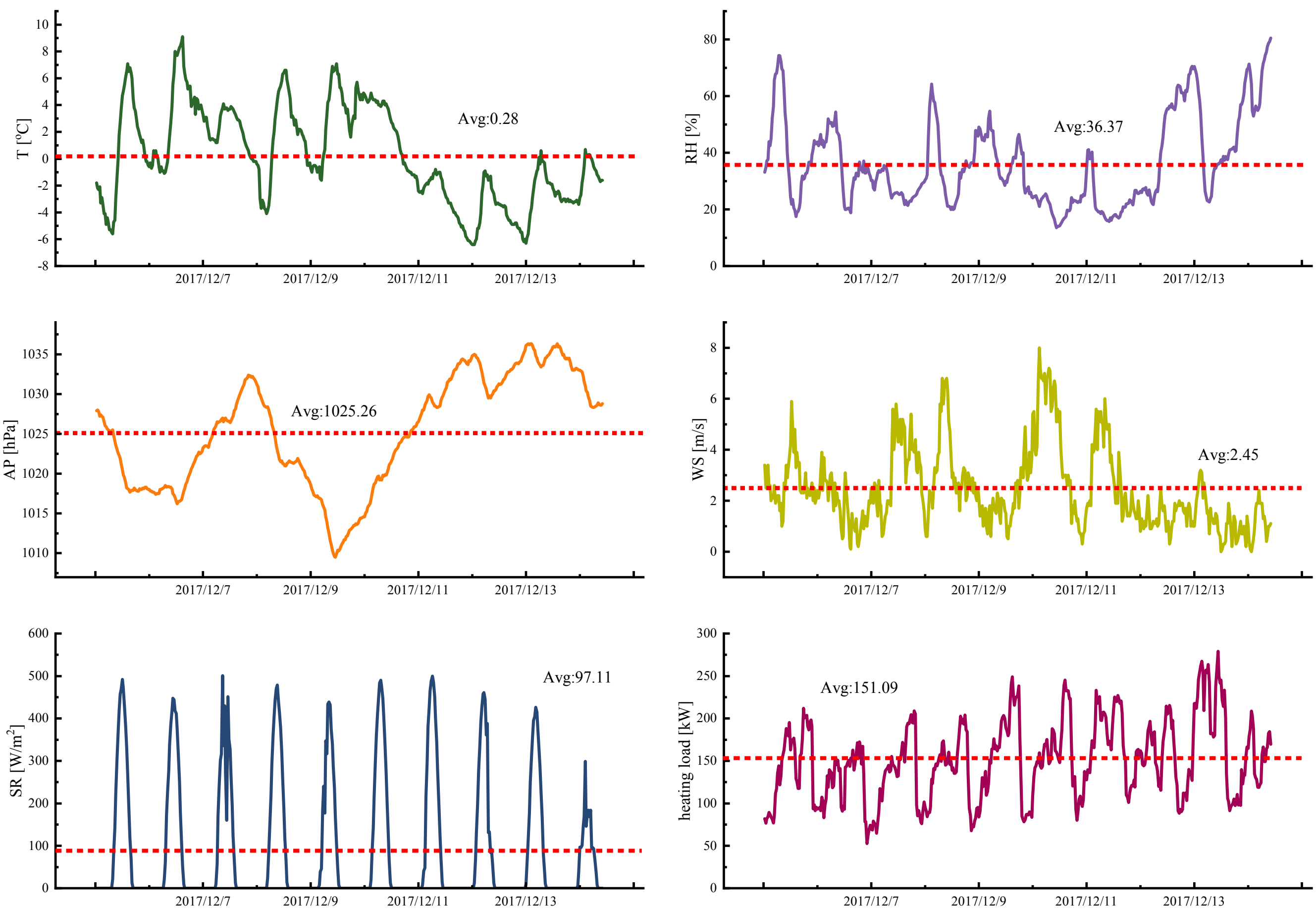
Table 5 summarizes the evaluation indicators of generalization performance. Fig. 10 shows the relative error fluctuations for each model, and the x-axis represents the number of each sample.

According to Fig. 10, the stacking model has the smallest generalization error fluctuation, with accuracy improvement achieving about 6.7%–29.5% for Case A and 7.1%-34.6% for Case B, in comparison to the other five models. According to Table 5, the ranking of the generalization abilities of the six models reflected by the four indicators is consistent. The CVRMSE, RMSE, MAE, and MAPE of the stacking model are 12.86%, 28.69, 20.88, and 9.90%, respectively, for Case A, and 12.79%, 18.40, 13.98, and 10.70%, respectively, for Case B. Each of these indicators is lower than it is in the other five prediction models. For Case A, the kNN model has the worst generalization performance; although it has higher precision for the training dataset, the accuracy on the testing dataset is poor. For Case B, the kNN model performs relatively well. This indicates that the model is less stable for different cases. The GBDT model has superior generalization performance for both cases. The SVR model's generalization for Case B is better than for Case A, compared to other models—basically the same as that of the GBDT model. Overall, the stacking model has better generalization performance compared to the other five models.

5.3. Model robustness

Multiple testing datasets with different noise intensities can be formed by adding Gaussian distribution white noise to different number samples randomly selected from the original testing datasets. There are four levels of noise intensity: 20%, 40%, 60%, and 80%. The stacking model and five base models were re-run on datasets of cases A and B, using four noise-introduced input profiles.

The variation of R^2 with noise intensities is used as an indicator to reveal model robustness. Fig. 11 shows the R^2 of various models at different noise intensities and reveals that R^2 has a decreasing trend as the noise intensity increases. For the stacking model, when the noise intensity reaches 80%, the R^2 of the stacking model changes from 0.825 to 0.795 (Case A) and from 0.830 to 0.789 (Case B). The kNN and SVR models result in the worst performance. From the perspective of the attenuation rate, the performance of the RF model is the best, especially for Case A; R^2 is only changed from 0.779 to 0.776 when the noise level is increased to 80%. Although the attenuation rate of the stacking model is not minimal—in fact, it is greater than that of the RF, GBDT, and XGBoost models for Case A and greater than that of the RF model for Case B—the absolute performance is optimal at each noise intensity based on R^2. In addition, the average R^2 of the model under different noise intensities is also optimal, with the promotion rate being 1.5%–34.1% for Case A and 1.8%–19.3% for Case B, compared to other models.



(b)

Fig. 5. (*continued*)

5.4. Generalizability of the proposed method

Some existing studies have confirmed the effectiveness of using machine learning algorithms to build meta-models of building performance in early building performance evaluation [74]. Similar to the SVR and ANN algorithms, the proposed stacking model also can be used to build meta-models. The biggest difference between building energy prediction and meta-model problems is model inputs. The input of the former usually involves some variables that affect the actual building energy consumption, including the outdoor temperature, relative humidity, etc. It can be obtained through actual testing. The input of the latter usually relates to some building design parameters, which can be obtained via a sampling algorithm. By using the meta-model built by the stacking algorithm, designers can quickly predict energy building performance based on a set of design options.

The meta-model established by the proposed stacking algorithm can be further extended to the problem of building performance optimization. In building performance optimization design, especially for multi-objective problems, meta-models based optimization algorithm is often performed to solve the optimization model [83]. At present, algorithms such as SVR, ANN, and RF have been involved in many building performance optimizations [84]. They can be adopted to build meta-models of building design parameters and performance indicators. Then, meta-models can treat as the fitness function of the optimization algorithm to participate in the optimization process, avoiding long calculation times caused by invoking simulation software.

Fig. 6. Collection of operational data.

Table 2
Monitoring instrument and accuracy.

Name (type)	Measured Parameters	Measured Accuracy	Measured Range
Portable ultrasonic flowmeter (TDS-100P)	Velocity	≥1%	–
Dual probe (TR004)	Water temperature	± 0.5 °C	−30 ~ 125 °C
Weather station (Onset-U30)	Outdoor air temperatures	± 0.2 °C	−40–75 °C
	Relative humidity levels	± 2.5%	0–100%
	Global solar radiation levels	± 10 W/m^2	0–1280 W/m^2

Table 3
Optimized hyperparameters of all models.

Model	Parameter	Case A	Case B
RF	MD	10	3
	MF	15	9
	NT	1000	100
XGBoost	MD	2	5
	NT	1000	100
	LR	0.1	0.5
	subsample	0.9	0.9
GBDT	MD	3	5
	MF	10	9
	NT	500	100
	LR	0.01	0.05
SVR	c	100	100
	γ	0.1	1
kNN	k	3	5
	p	2	2

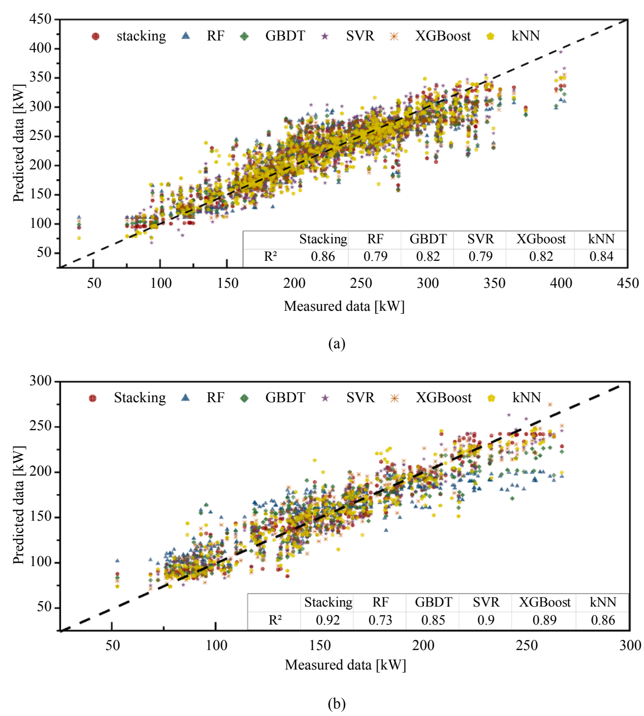


Fig. 7. The fitting characteristics of measured and predicted data: (a) Case A; (b) Case B.

6. Conclusions

Even as building energy consumption prediction becomes more critical in building energy management systems, it is still a challenge to continuously improve the performance of prediction models in conjunction with engineering applications. Based on the idea of model integration, this paper has proposed a novel model (the stacking model) for building energy consumption prediction.

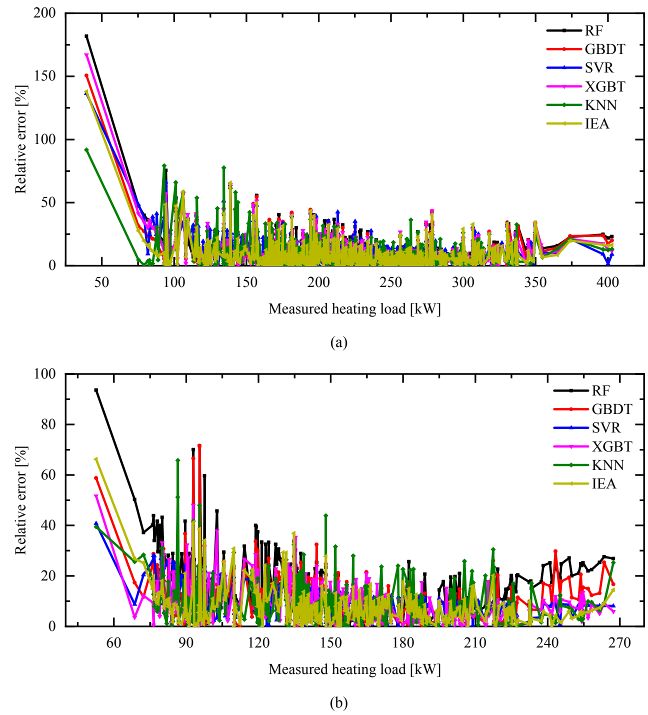


Fig. 8. The distribution of relative errors: (a) Case A; (b) Case B.

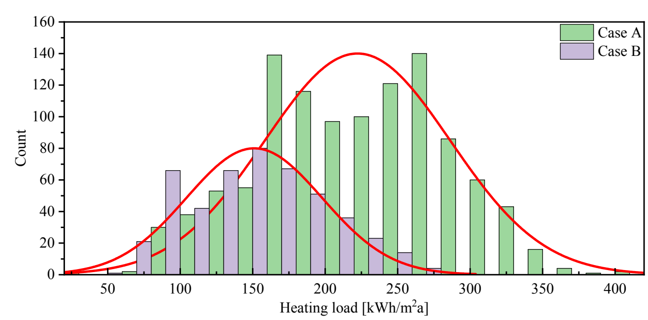


Fig. 9. The distribution of heating load in two cases.

Table 4
Accuracy metrics of different models.

		Stacking	RF	GBDT	SVR	XGBoost	kNN
Case A	CVRMSE (%)	10.60	12.93	12.09	12.76	12.02	11.28
	RMSE (kW)	23.53	28.70	26.83	28.33	26.67	25.03
	MAE (kW)	16.14	20.05	18.41	22.67	19.11	17.30
	MAPE (%)	7.66	9.71	8.72	11.20	9.30	8.47
Case B	CVRMSE (%)	8.96	17.08	12.45	9.97	10.45	11.55
	RMSE (kW)	13.81	26.34	19.20	15.37	16.12	17.81
	MAE (kW)	10.61	21.21	14.73	13.52	12.99	12.89
	MAPE (%)	7.51	14.84	10.21	9.64	9.31	8.96

Table 5
Generalization performances of different models.

Case	Indicators	Stacking	RF	GBDT	SVR	XGBoost	kNN
Case A	CVRMSE (%)	12.86	14.40	13.76	15.28	13.87	17.55
	RMSE (kW)	28.69	32.13	30.69	34.09	30.94	39.14
	MAE (kW)	20.88	23.32	22.40	26.36	22.84	27.11
	MAPE (%)	9.90	11.07	10.61	13.10	11.04	14.03
Case B	CVRMSE (%)	12.79	17.27	13.71	13.50	15.55	15.32
	RMSE (kW)	18.40	24.85	19.73	19.42	22.38	22.04
	MAE (kW)	13.98	20.05	15.10	14.83	17.62	16.39
	MAPE (%)	10.70	16.35	11.87	11.51	13.54	12.76

The core idea of the stacking model is to collect the differentiation of various base algorithms (each of which can observe data from different spatial and structural perspectives) by constructing an integrated framework. Based on popularity and diversity, several algorithms, including the RF, GBDT, XGBoost, SVR, and kNN models, are selected as base models in the first layer. The stacking model develops each base model via fivefold cross-validation and fuses outputs of base models in the form of "meta-features". The application results in two real campus buildings show that the proposed model can provide more accurate energy consumption than the other models.

The results of model accuracy are as follows:

- The stacking model has higher accuracy than any of the other five base models, showing a CVRMSE, RMSE, MAE, and MAPE of 10.60%, 23.53, 16.14, and 7.66%, respectively, for Case A, and 8.96%, 13.81, 10.61 and 7.51%, respectively, for Case B. The stacking model has the best fit between predicted and measured data, with R^2 equal to 0.86 for Case A and 0.92 for Case B. Additionally, the stacking model exhibits superior characteristics when the heating load is in the middle position; although it is not optimal with lower heating load, which matches the distribution of the actual heating load of "more in the middle and less at both

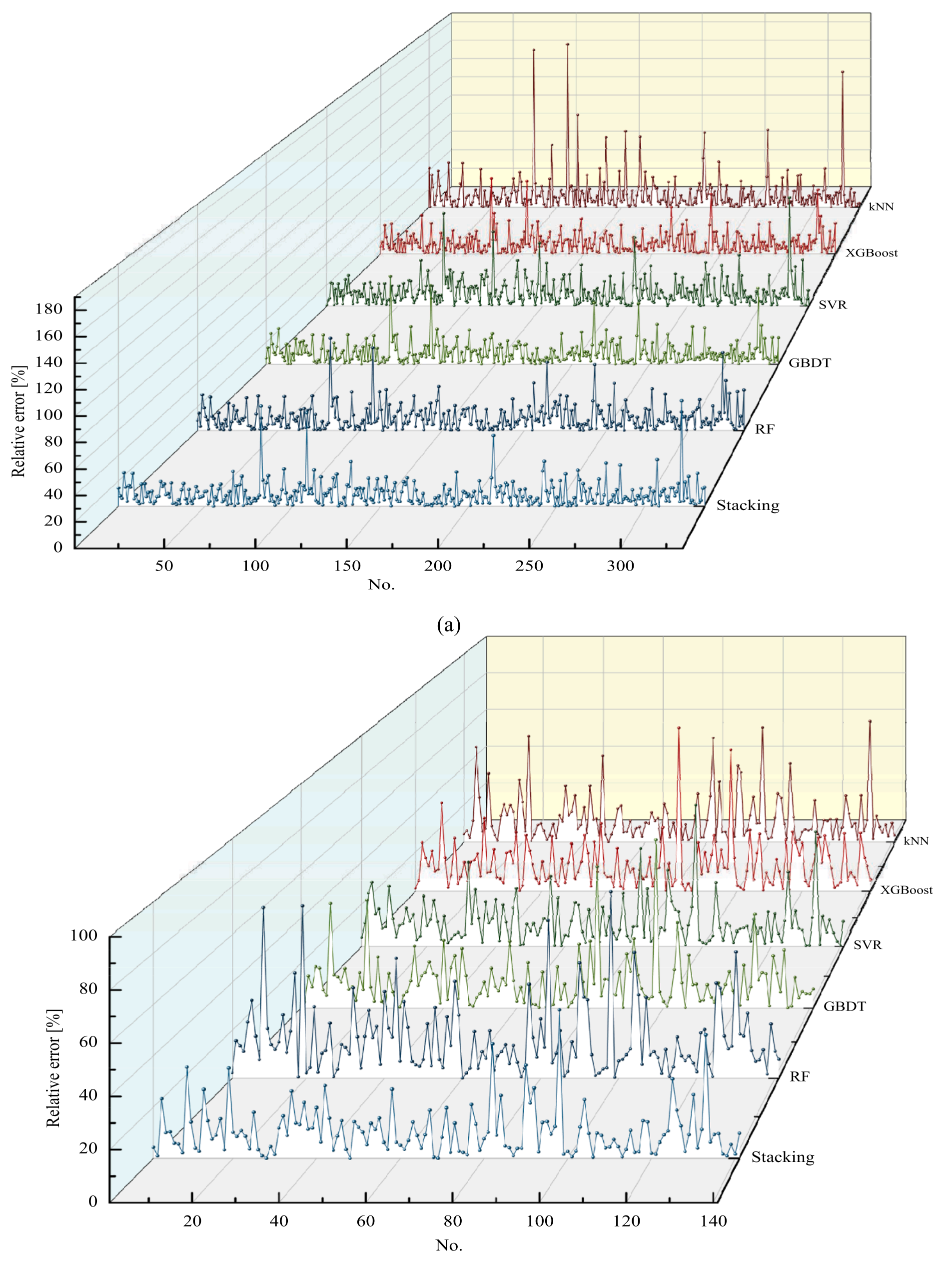
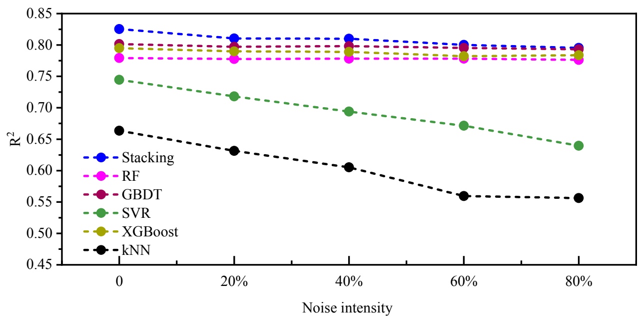
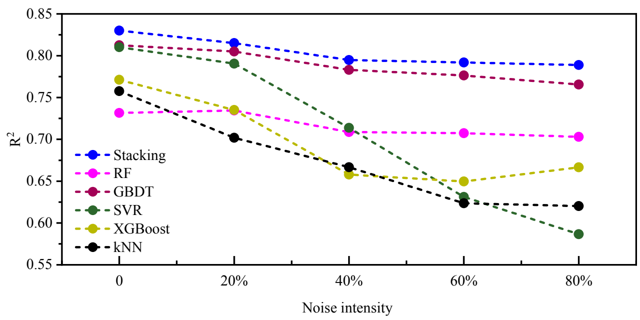


Fig. 10. Fluctuations in relative errors: (a) Case A; (b) Case B.



(a)



(b)

Fig. 11. Accuracy reduction of different models under various noise: (a) Case A; (b) Case B.

ends." When based on the mean relative error, the stacking method achieves higher accuracy than any of the other five models, with accuracy improvement being about 9.5%–31.6% for Case A and 16.2%–49.4% for Case B. Therefore, from the perspective of engineering practice, the improved integrated algorithm has better practicability.

Furthermore, model performance is evaluated from the perspectives of generalization and robustness by comparing it with five base models. The main conclusions are as follows:

- From the aspect of generalization, the stacking model performs best. The CVRMSE, RMSE, MAE, and MAPE are 12.86%, 28.69, 20.88, and 9.90% for Case A, and 12.79%, 18.40, 13.98, and 10.70% for Case B. When based on the mean relative generalization error, it achieves an accuracy improvement of about 6.7%–29.5% for Case A and 7.1%-34.6% for Case B, compared to other models.
- From the aspect of robustness, the absolute performance of the stacking model is optimal at different noise intensities, although the attenuation rate is not the lowest. When the noise intensity reaches 80%, the R^2 of the stacking model changes from 0.825 to 0.795 (Case A) and from 0.830 to 0.789 (Case B). The promotion rate of the average R^2 of the model under different noise intensities reaches 1.5%–34.1% for Case A and 1.8%–19.3% for Case B, compared to other models.

The proposed model can enrich empirical model databases of building energy consumption prediction and improves the overall performance of energy consumption prediction through actual case verification.

CRediT authorship contribution statement

Ran Wang: Conceptualization, Data curation, Formal analysis, Writing - original draft, Writing - review & editing. **Shilei Lu:** Funding acquisition, Methodology, Project administration, Supervision. **Wei Feng:** Validation, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research has been supported by the "National Key R&D Program of China" (Grant No. 2016YFC0700100).

References

[1] IEA. Energy Efficiency: Buildings. https://wwwieaorg/topics/energyefficiency/buildings/; 2018.

[2] Hong T, Koo C, Kim J, Lee M, Jeong K. A review on sustainable construction management strategies for monitoring, diagnosing, and retrofitting the building's dynamic energy performance: Focused on the operation and maintenance phase. Appl Energy 2015;155:671–707.

[3] O'Dwyer E, Pan I, Acha S, Shah N. Smart energy systems for sustainable smart cities: Current developments, trends and future directions. Appl Energy 2019;237:581–97.

[4] Wang Z, Hong T, Piette MA. Data fusion in predicting internal heat gains for office buildings through a deep learning approach. Appl Energy 2019;240:386–98.

[5] Wang L, Lee EW, Yuen RK. Novel dynamic forecasting model for building cooling loads combining an artificial neural network and an ensemble approach. Appl Energy 2018;228:1740–53.

[6] Dahl M, Brun A, Andresen GB. Using ensemble weather predictions in district heating operation and load forecasting. Appl Energy 2017;193:455–65.

[7] Bedi J, Toshniwal D. Deep learning framework to forecast electricity demand. Appl Energy 2019;238:1312–26.

[8] Fan C, Xiao F, Wang S. Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques. Appl Energy 2014;127:1–10.

[9] Afroz Z, Urmee T, Shafiullah G, Higgins G. Real-time prediction model for indoor temperature in a commercial building. Appl Energy 2018;231:29–53.

[10] Cui B, Fan C, Munk J, Mao N, Xiao F, Dong J, et al. A hybrid building thermal modeling approach for predicting temperatures in typical, detached, two-story houses. Appl Energy 2019;236:101–16.

[11] von Grabe J. Potential of artificial neural networks to predict thermal sensation votes. Appl Energy 2016;161:412–24.

[12] Lu S, Li Q, Bai L, Wang R. Performance predictions of ground source heat pump system based on random forest and back propagation neural network models. Energ Convers Manage 2019;197:111864.

[13] Bianchini G, Casini M, Pepe D, Vicino A, Zanvettor GG. An integrated model predictive control approach for optimal HVAC and energy storage operation in large-scale buildings. Appl Energy 2019;240:327–40.

[14] Hu R, Granderson J, Auslander D, Agogino A. Design of machine learning models with domain experts for automated sensor selection for energy fault detection. Appl Energy 2019;235:117–28.

[15] Zhong H, Wang J, Jia H, Mu Y, Lv S. Vector field-based support vector regression for building energy consumption prediction. Appl Energy 2019;242:403–14.

[16] Fan C, Ding Y. Cooling load prediction and optimal operation of HVAC systems using a multiple nonlinear regression model. Energy Build 2019;197:7–17.

[17] Min Y, Chen Y, Yang H. A statistical modeling approach on the performance prediction of indirect evaporative cooling energy recovery systems. Appl Energy 2019;255:113832.

[18] Joe J, Karava P. A model predictive control strategy to optimize the performance of radiant floor heating and cooling systems in office buildings. Appl Energy 2019;245:65–77.

[19] Zhao Y, Li T, Zhang X, Zhang C. Artificial intelligence-based fault detection and diagnosis methods for building energy systems: Advantages, challenges and the future. Renew Sustain Energy Rev 2019;109:85–101.

[20] Cai M, Pipattanasomporn M, Rahman S. Day-ahead building-level load forecasts using deep learning vs. traditional time-series techniques. Appl Energy 2019;236:1078–88.

[21] Wei Y, Zhang X, Shi Y, Xia L, Pan S, Wu J, et al. A review of data-driven approaches for prediction and classification of building energy consumption. Renew Sustain Energy Rev 2018;82:1027–47.

[22] Shine P, Scully T, Upton J, Murphy M. Annual electricity consumption prediction and future expansion analysis on dairy farms using a support vector machine. Appl Energy 2019;250:1110–9.

[23] Chaudhuri T, Soh YC, Li H, Xie L. A feedforward neural network based indoor-climate control framework for thermal comfort and energy saving in buildings. Appl Energy 2019;248:44–53.

[24] Fang T, Lahdelma R. Evaluation of a multiple linear regression model and SARIMA model in forecasting heat demand for district heating system. Appl Energy 2016;179:544–52.

[25] Wang Z, Srinivasan R. A review of artificial intelligence based building energy use prediction: Contrasting the capabilities of single and ensemble prediction models. Renew Sustain Energy Rev 2017;75:796–808.

[26] Yuan T, Zhu N, Shi Y, Chang C, Yang K, Ding Y, et al. Sample data selection method for improving the prediction accuracy of the heating energy consumption. Energy

Build 2018.
[27] Singh P, Dwivedi P. Integration of new evolutionary approach with artificial neural network for solving short term load forecast problem. Appl Energy 2018;217:537–49.
[28] Chae YT, Horesh R, Hwang Y, Lee YM. Artificial neural network model for forecasting sub-hourly electricity usage in commercial buildings. Energy Build 2016;111:184–94.
[29] Jetcheva JG, Majidpour M, Chen W-P. Neural network model ensembles for building-level electricity load forecasts. Energy Build 2014;84:214–23.
[30] Jung HC, Kim JS, Heo H. Prediction of building energy consumption using an improved real coded genetic algorithm based least squares support vector machine approach. Energy Build 2015;90:76–84.
[31] Chitsaz H, Shaker H, Zareipour H, Wood D, Amjady N. Short-term electricity load forecasting of buildings in microgrids. Energy Build 2015;99:50–60.
[32] Edwards RE, New J, Parker LE. Predicting future hourly residential electrical consumption: A machine learning case study. Energy Build 2012;49:591–603.
[33] Escrivá-Escrivá G, Álvarez-Bel C, Roldán-Blay C, Alcázar-Ortega M. New artificial neural network prediction method for electrical consumption forecasting based on building end-uses. Energy Build 2011;43:3112–9.
[34] Yezioro A, Dong B, Leite F. An applied artificial intelligence approach towards assessing building performance simulation tools. Energy Build 2008;40:612–20.
[35] Leung M, Norman C, Lai LL, Chow TT. The use of occupancy space electrical power demand in building cooling load prediction. Energy Build 2012;55:151–63.
[36] Ben-Nakhi AE, Mahmoud MA. Cooling load prediction for buildings using general regression neural networks. Energ Convers Manage 2004;45:2127–41.
[37] Li Q, Meng Q, Cai J, Yoshino H, Mochida A. Applying support vector machine to predict hourly cooling load in the building. Appl Energy 2009;86:2249–56.
[38] Amasyali K, El-Gohary NM. A review of data-driven building energy consumption prediction studies. Renew Sustain Energy Rev 2018;81:1192–205.
[39] Zhao H-x, Magoulès F. A review on the prediction of building energy consumption. Renew Sustain Energy Rev 2012;16:3586–92.
[40] Heinemann G, Nordmian D, Plant E. The relationship between summer weather and summer loads-a regression analysis. IEEE T Power Syst 1966:1144–54.
[41] Apadula F, Bassini A, Elli A, Scapin S. Relationships between meteorological variables and monthly electricity demand. Appl Energy 2012;98:346–56.
[42] Christiaanse W. Short-term load forecasting using general exponential smoothing. IEEE T Power Syst 1971:900–11.
[43] Amjady N. Short-term hourly load forecasting using time-series modeling with peak load estimation capability. IEEE T Power Syst 2001;16:498–505.
[44] Becker R, Thrän D. Completion of wind turbine data sets for wind integration studies applying random forests and k-nearest neighbors. Appl Energy 2017;208:252–62.
[45] Burger EM, Moura SJ. Gated ensemble learning method for demand-side electricity load forecasting. Energy Build 2015;109:23–34.
[46] Long H, Zhang Z, Su Y. Analysis of daily solar power prediction with data-driven approaches. Appl Energy 2014;126:29–37.
[47] Ahmad A, Hassan M, Abdullah M, Rahman H, Hussin F, Abdullah H, et al. A review on applications of ANN and SVM for building electrical energy consumption forecasting. Renew Sustain Energy Rev 2014;33:102–9.
[48] Chen Y, Xu P, Chu Y, Li W, Wu Y, Ni L, et al. Short-term electrical load forecasting using the Support Vector Regression (SVR) model to calculate the demand response baseline for office buildings. Appl Energy 2017;195:659–70.
[49] Dong B, Cao C, Lee SE. Applying support vector machines to predict building energy consumption in tropical region. Energy Build 2005;37:545–53.
[50] Ekici BB, Aksoy UT. Prediction of building energy consumption by using artificial neural networks. Adv Eng Softw 2009;40:356–62.
[51] Reid SJDoCS, University of Colorado at Boulder. A review of heterogeneous ensemble methods; 2007.
[52] Chou J-S, Bui D-KJE. Modeling heating and cooling loads by artificial intelligence for energy-efficient building design. Buildings 2014; 82: 437–46.
[53] Wang Z, Wang Y, Zeng R, Srinivasan RS, Ahrentzen S. Random Forest based Hourly Building Energy Prediction. Energy Build 2018;171.
[54] Tsanas A, Xifara A. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. Energy Build 2012;49:560–7.
[55] Candanedo LM, Feldheim V, Deramaix D. Data driven prediction models of energy use of appliances in a low-energy house. Energy Build 2017;140.
[56] Touzani S, Granderson J, Fernandes S. Gradient boosting machine for modeling the energy consumption of commercial buildings. Energy Build 2018;158:1533–43.
[57] Ahmad T, Chen H. Nonlinear autoregressive and random forest approaches to forecasting electricity load for utility energy management systems. Sustain Cities Soc 2019;45:460–73.
[58] Chakraborty D, Elzarka H. Early detection of faults in HVAC systems using an XGBoost model with a dynamic threshold. Energy Build 2019;185:326–44.
[59] Papadopoulos S, Kontokosta CE. Grading buildings on energy performance using city benchmarking data. Appl Energy 2019;233:244–53.
[60] Ding Y, Zhang Q, Yuan T, Yang F. Effect of input variables on cooling load prediction accuracy of an office building. Appl Therm Eng 2018;128.
[61] Ding Y, Zhang Q, Yuan T. Research on short-term and ultra-short-term cooling load prediction models for office buildings. Energy Build 2017;154.
[62] Chen Y, Tan H. Short-term prediction of electric demand in building sector via hybrid support vector regression. Appl Energy 2017;204:1363–74.
[63] Li K, Hu C, Liu G, Xue W. Building's electricity consumption prediction using optimized artificial neural networks and principal component analysis. Energy Build 2015;108:106–13.
[64] Fan C, Xiao F, Zhao Y. A short-term building cooling load prediction method using deep learning algorithms. Appl Energy 2017;195:222–33.
[65] Brusaferri A, Matteucci M, Portolani P, Vitali A. Bayesian deep learning based method for probabilistic forecast of day-ahead electricity prices. Appl Energy 2019;250:1158–75.
[66] Li Q, Meng Q, Cai J, 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.
[67] Massana J, Pous C, Burgas L, Melendez J, Colomer JJE. Buildings. Short-term load forecasting in a non-residential building contrasting models and attributes. Energy Build 2015;92:322–30.
[68] Wang Z, Srinivasan RS, Shi J. Artificial intelligent models for improved prediction of residential space heating. J Energ Eng 2016;142:04016006.
[69] Farzana S, Liu M, Baldwin A, Hossain MU. Multi-model prediction and simulation of residential building energy in urban areas of Chongqing, South West China. Energy Build 2014;81:161–9.
[70] Zhang Y, O'Neill Z, Dong B, Augenbroe G. Comparisons of inverse modeling approaches for predicting building energy performance. Build Environ 2015;86:177–90.
[71] Jovanović RŽ, Sretenović AA, Živković BD. Ensemble of various neural networks for prediction of heating energy consumption. Energy Build 2015;94:189–99.
[72] Ahmad MWMM, Rezgui Y. Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption. Energy Build 2017;147:77–89.
[73] Wang R, Lu S, Li Q. Multi-criteria comprehensive study on predictive algorithm of hourly heating energy consumption for residential buildings. Sustain Cities Soc 2019:101623.
[74] Ostergard T, Jensen RL, Maagaard SE. A comparison of six metamodeling techniques applied to building performance simulations. Appl Energy 2018;211:89–103.
[75] Fan C, Wang J, Gang W, Li S. Assessment of deep recurrent neural network-based strategies for short-term building energy predictions. Appl Energy 2019;236:700–10.
[76] Breiman L. Random forests. Mach Learn 2001;45:5–32.
[77] Jiang R, Tang W, Wu X, Fu W. A random forest approach to the detection of epistatic interactions in case-control studies. BMC Bioinf 2009;10:S65.
[78] Keller JM, Gray MR, Givens JA. A fuzzy k-nearest neighbor algorithm. IEEE T Power Syst 1985:580–5.
[79] Drucker H, Burges CJ, Kaufman L, Smola AJ, Vapnik V. Support vector regression machines. Adv Neural Inform Process Syst 1997. p. 155–61.
[80] Friedman JH. Greedy function approximation: a gradient boosting machine. Ann Stat 2001;29:1189–232.
[81] Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, et al. Lightgbm: A highly efficient gradient boosting decision tree. Adv Neural Inform Process Syst 2017:3146–54.
[82] Chen T, Guestrin C. Xgboost: A scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining: ACM; 2016. p. 785-94.
[83] Wang R, Lu S, Feng W. A three-stage optimization methodology for envelope design of passive house considering energy demand, thermal comfort and cost. Energy 2019;116723.
[84] Westermann P, Evins R. Surrogate modelling for sustainable building design-A review. Energy Build 2019.