Energy & Buildings 208 (2020) 109675

Contents lists available at ScienceDirect

Energy & Buildings

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

Study on deep reinforcement learning techniques for building energy consumption forecasting

Tao Liu[a], Zehan Tan[b], Chengliang Xu[a], Huanxin Chen[a,*], Zhengfei Li[a]

[a] Department of Refrigeration and Cryogenic, Huazhong University of Science and Technology, Wuhan, China
[b] State Key Laboratory of Air-Conditioning Equipment and System Energy Conservation, Zhuhai, China

ARTICLE INFO

Article history:
Received 9 August 2019
Revised 29 October 2019
Accepted 2 December 2019
Available online 3 December 2019

Keywords:
Energy consumption prediction
Ground source heat pump
Deep reinforcement learning
Asynchronous advantage Actor-Critic
Deep deterministic Policy gradient
Recurrent deterministic Policy gradient

ABSTRACT

Reliable and accurate building energy consumption prediction is becoming increasingly pivotal in building energy management. Currently, data-driven approach has shown promising performances and gained lots of research attention due to its efficiency and flexibility. As a combination of reinforcement learning and deep learning, deep reinforcement learning (DRL) techniques are expected to solve nonlinear and complex issues. However, very little is known about DRL techniques in forecasting building energy consumption. Therefore, this paper presents a case study of an office building using three commonly-used DRL techniques to forecast building energy consumption, namely Asynchronous Advantage Actor-Critic (A3C), Deep Deterministic Policy Gradient (DDPG) and Recurrent Deterministic Policy Gradient (RDPG). The objective is to investigate the potential of DRL techniques in building energy consumption prediction field. A comprehensive comparison between DRL models and common supervised models is also provided.

The results demonstrate that the proposed DDPG and RDPG models have obvious advantages in forecasting building energy consumption compared to common supervised models, while accounting for more computation time for model training. Their prediction performances measured by mean absolute error (MAE) can be improved by 16%-24% for single-step ahead prediction, and 19%-32% for multi-step ahead prediction. The results also indicate that A3C performs poor prediction accuracy and shows much slower convergence speed than DDPG and RDPG. However, A3C is still the most efficient technique among these three DRL methods. The findings are enlightening and the proposed DRL methodologies can be positively extended to other prediction problems, e.g., wind speed prediction and electricity load prediction.

1. Introduction

Building sector has become the largest energy consumer worldwide due to population growth, increases of people comfort demands, and global climate change [1]. Specifically, buildings are responsible for approximately 32% of world's total energy consumption and even 40% in many developed countries (e.g., around 39% in the U.S. and 40% in Europe) [2,3]. Besides, new policies and regulations have been promulgated in many countries for the effective design of new buildings, with the aim of achieving building energy conservation. Hence, improving building energy efficiency has become a paramount issue around the world. In this context, reliable and accurate prediction of building energy consumption is becoming increasing favorable and vital in improving building energy efficiency, since it plays a fundamental role in many building energy management tasks, e.g., system fault detection and diagnosis [4], optimal operation strategy control [5], and demand-side management [6]. According to Ref. [7], it was shown that building energy savings can reach 10% to 30% with reliable energy consumption predictions, indicating the great significance of building energy consumption prediction for building energy efficiency improvement.

Broadly speaking, existing approaches for building energy consumption prediction can be classified into three categories, i.e., engineering approach, statistical approach and artificial intelligence (AI) based approach [8]. Engineering methods, aka white-box methods, rely on the elaborate physical functions and thermodynamic rules, and require developing and solving many physical equations for building energy behaviors estimation. Besides, a large number of building parameters are needed for engineering calculation, such as building construction details, thermal properties of building material, weather condition and building occupancy, which are not always available. For these

* Corresponding author.
E-mail address: chenhuanxin@tsinghua.org.cn (H. Chen).

https://doi.org/10.1016/j.enbuild.2019.109675
0378-7788/

Nomenclature

RL	Reinforcement learning
DRL	Deep reinforcement learning
A3C	Asynchronous Advantage Actor-Critic
DDPG	Deep Deterministic Policy Gradient
RDPG	Recurrent Deterministic Policy Gradient
LOF	Local outlier factor
SVM	Support vector machine
MLR	Multiple linear regression
ANN	Artificial neuron network
DT	Decision tree
BPNN	Back-propagation neural network
MLP	Multi-layer perceptron
ACF	Autocorrelation Function
PACF	Partial Autocorrelation Function
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
RF	Random Forest
CART	Classification and Regression Trees
MAE	mean absolute error
RMSE	root mean square error
CV	coefficient of variance
R^2	coefficient of
GSHP	determination
EC	Ground source heat pump
OT	Energy consumption
WS	Outdoor temperature
RH	Wind speed
SS	Relative humidity
s	System status
a	state
r	action
π	reward
$V(s)$	policy
$Q(s, a)$	Value function
γ	Action-value function
θ	Discount factor
w	Parameters of Actor network
δ	Parameters of Critic network
$\nabla_\theta \log \pi_\theta(s, a)$	TD-error
α	Score function
β	Learning rate of Actor Learning rate of Critic

reasons, the accurate estimation of building energy consumption based on engineering methods is difficult and time-consuming, and has intrinsic limitations in practice [9]. In term of statistical approaches, they simply correlate building energy consumption with relevant input variables (e.g. weather variables), and have been identified some deficiencies in practice, of which the most important one is the lack of accuracy and flexibility [10].

By contrast, AI-based approaches work in a purely data-driven fashion and require little domain knowledge. Such data-driven methods can learn from historic data and aim to forecast the energy consumption based on previous energy use patterns. Specifically, they attempt to develop prediction models in a supervised manner through discovering and generalizing the underlying linear or nonlinear relationship between given inputs (e.g. historic energy data and meteorological data) and outputs (i.e. building energy consumption). On the other hand, massive data available from Building Automation System as well as the rapid development of data science make the establishment of data-driven models more convenient. Accordingly, data-driven methods have become a research hotspot in recent years due to their flexibility and efficiency compared to engineering and statistical methods [11].

The prediction performances of data-driven models are greatly influenced by three factors, i.e., the quality of recorded building energy consumption data, selection of the input variables and prediction algorithms for models development [12]. For the first one, many anomaly detection methods have been proposed and extensively used to remove outliers in raw data, including statistical-based anomaly detection methods (e.g. "3-sigma" principle and interquartile range rule) [13,14], density-based methods (e.g. LOF method) [15], and machine learning methods (e.g. one-class SVM and Isolation Forest) [16,17]. Regrading to input variables selection, previous studies mainly focus on engineering, statistical and structural features extraction [18–20]. With the development of deep learning, auto-encoder has become another powerful and popular feature extractor to select suitable variables as model inputs [21]. The last factor, i.e. the prediction algorithm utilized for model development, is part and parcel in data-driven models establishment process. In response, researchers have put their great effort on developing more robust models with higher accuracy and lower computation load. Supervised machine learning algorithms are the most widely used methods in forecasting building energy consumption, which can be classified into two categories, i.e., traditional machine learning methods and deep learning methods [22].

Traditional machine learning library mainly contains Multiple Linear Regression (MLR), Artificial Neural Network (ANN), Support Vector Machine (SVM), Decision Tree (DT) and their developments. Authors in Ref. [23] applied ANN approach to forecast the energy consumption of an administration building, and it was found that ANN based model can yield better prediction results when compared with simulation software prediction results. Li et al. [24] used SVM to predict hourly building cooling load, and the results showed that the forecasting performance of SVM was better than that of back-propagation neural network (BPNN). Yu et al. [25] employed DT algorithm to predict building energy demand levels, and the resulting prediction accuracy can reach 92% in testing data. To improve the robustness of models, ensemble learning algorithms were proposed. In Ref. [26], data decomposition based ensemble models are investigated for ground source heat pump load forecasting. The prediction results showed the ensemble models could evidently enhance prediction accuracy. Wang et al. [9] proposed Ensemble Bagging Trees (EBT) to predict building hourly electricity demand. In this ensemble model, DT was used as the base model and the EBT model output its prediction results by averaging the outputs of each DT base models. The results indicated that this proposed ensemble model was superior to single prediction model in accuracy and stability. Nevertheless, establishing such ensemble models were time-consuming and needed more computation load.

The above-mentioned prediction methods, usually adopt 'shallow' structures for modeling, which lead to limited power in features extraction of their raw inputs. Deep learning, as the evolution of ANN, has multiple processing layers to automatically learn suitable representations of raw inputs, thereby overcoming the intrinsic deficiency of traditional machine learning algorithms [27]. In the field of building energy consumption prediction, deep learning has also gained a lot of research attention. In Ref. [28], authors predicted monthly building energy consumption using three deep learning algorithms, including Deep Full Connected, Convolutional and Long Short-Term Memory neural networks. Fu [29] adopted deep brief network combined with ensemble technique for building cooling load forecasting, and obtained competitive accuracy. Rahman [30] developed and optimized deep Recurrent Neural Network (RNN) models, to make medium-to-long term building electricity consumption, and found RNN outperformed 3-layered

perception neural network. Some other similar studies can be seen in Ref. [11,22,31].

Deep learning has gained the huge success not only in building energy consumption prediction, but also in many other areas, such as visual object recognition and speech recognition [27]. Deep reinforcement learning (DRL), which is a sub-family of deep learning algorithms, is an active area in the artificial intelligence community. DRL integrates the perceptual ability of deep learning and the decision-making ability of reinforcement learning, thereby realizing the direct control for complicated control problems with high-dimensional action space. DRL has made numerous breakthroughs in various fields, such as games [32], robotics [33], as well as smart driving [34]. In the building field, some recent works have developed DRL based models for optimal control of a variety of building systems, and good performances were achieved [35–37]. However, such a promising technique, i.e. DRL, has rarely deployed for building energy consumption prediction, and its potential in forecasting building energy consumption is still unknown.

To fill this research gap, this paper systematically investigates the potential of DRL algorithms in forecasting building energy consumption. Three commonly-used DRL models, including Asynchronous Advantage Actor-Critic (A3C), Deep Deterministic Policy Gradient (DDPG) and Recurrent Deterministic Policy Gradient (RDPG), are established for building energy consumption prediction, and their comparative analysis is conducted from three perspectives, i.e., prediction accuracy, convergence speed, and computation time. In addition, the detailed comparisons between DRL models and supervised models are also given.

The rest of this paper is structured as follows. Section 2 introduces the research outline and the theoretical background of methodologies as well as evaluation indices utilized in this study. Section 3 describes the case building and data used. Data preparation process and modeling process are also presented in this part. In Section 4, the prediction results of DRL models and supervised models are presented and compared. Conclusions are drawn in Section 5.

2. Research methodology

2.1. Research outline

Fig. 1 illustrates the research outline of this paper. Firstly, the energy consumption data is collected from case building with a 5 min resolution. Meteorological data and expert knowledge are also introduced to enhance prediction accuracy and robustness. Then, dataset establishment and data preparation are conducted. Data preparation process mainly contains two tasks, i.e. outlier detection and feature extraction. For outlier detection, Local Outlier Factor (LOF) method is adopted to remove potential outliers from building daily energy consumption profiles. And for feature extraction, Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) are deployed to select the optimal lag period. Whereafter, three common supervised models along with three prevailing DRL models are developed based on same input variables. Finally, a rounded comparison about the performances of these six models is presented from three perspectives, i.e. prediction accuracy, convergence speed and computation time.

The following subsections present an overview on data preparation methods and prediction techniques. The evaluation indices used in this paper are also exhibited.

2.2. Data preparation methods

Data preparation methods mainly contain outlier detection method and feature extraction method. Outlier detection method is employed to remove the potential outliers in the raw data, thereby enhancing the data quality. In this work, LOF algorithm is applied to detect the abnormal data from building daily energy consumption profiles. LOF algorithm is a density-based unsupervised technique for identifying abnormal data and local outliers, which has been proved to be useful in previous studies [15,38,39]. LOF finds possible outliers by calculating local density deviation (defined as LOF value) for each sample to their neighbors. If a sample belongs to a dense cluster and is normal, it tends to have a low value of LOF as the average local reachability density of its neighborhoods is close to corresponding local reachability density of the sample. In contrast, samples deviating from the overall observations have higher values of LOF compared to normal points. That means, samples with higher LOF values, have more sparse neighborhoods and are more likely to be considered as outliers.

With respect to features extraction method, ACF and PACF are used to analyze the inherent correlation between observations in a time series. The ACF denotes the linear correlation between two time points of a variable, while the PACF denotes the correlation between the two time points without considering the effect of observations between them. Therefore, ACF and PACF can be utilized to choose appropriate input features for time series data and help determine potentially useful model structures.

2.3. Deep reinforcement learning

2.3.1. Reinforcement learning

Reinforcement learning (RL) is a sub-family of machine learning, which studies how the artificial agent performs the optimal action based on observed environment state by reward and punishment [40]. Almost all RL problems can be described as decision-making problems. There are five significant concepts in RL: state (denoted as s), action (denoted as a), reward (denoted as r), policy (denoted as π), as well as value function (denoted as $V(s)$) or action-value function (denoted as $Q(s, a)$). Here, policy defines the agent's behavior function. At time step t, agent firstly observes the environment state s_t, executes action a_t (according to policy π), and receives immediate reward r_t. The immediate reward is a scalar feedback signal which can indicate how well agent is doing and how far it is from the optimal policy (denoted as π^*). The final goal of RL is to find the optimal policy.

In a RL algorithm, either value function or action-value function is used for the prediction of future reward. Value function denotes the expected total discount reward starting from state s:

$$V(s) = E_\pi \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s \right] \tag{1}$$

where γ is discount factor. Action-value function denotes the expected total discount reward starting from state s and taking action a:

$$Q(s, a) = E \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s, a_t = a \right] \tag{2}$$

2.3.2. Actor-Critic

Currently prevailing deep reinforcement learning (DRL) algorithms, integrates the perceptual ability of deep learning and the decision-making ability of reinforcement learning, thereby realizing the direct control for complicated control problems with high-dimensional action space. DRL algorithms deploy non-linear approximator such as neural networks to estimate the value function (or action-value function) and current policy. A3C, DDPG as well as RDPG are three of the most commonly-used DRL techniques, which have yielded good performances in many continuous control tasks [41–43]. All these three DRL techniques are based on Actor-Critic

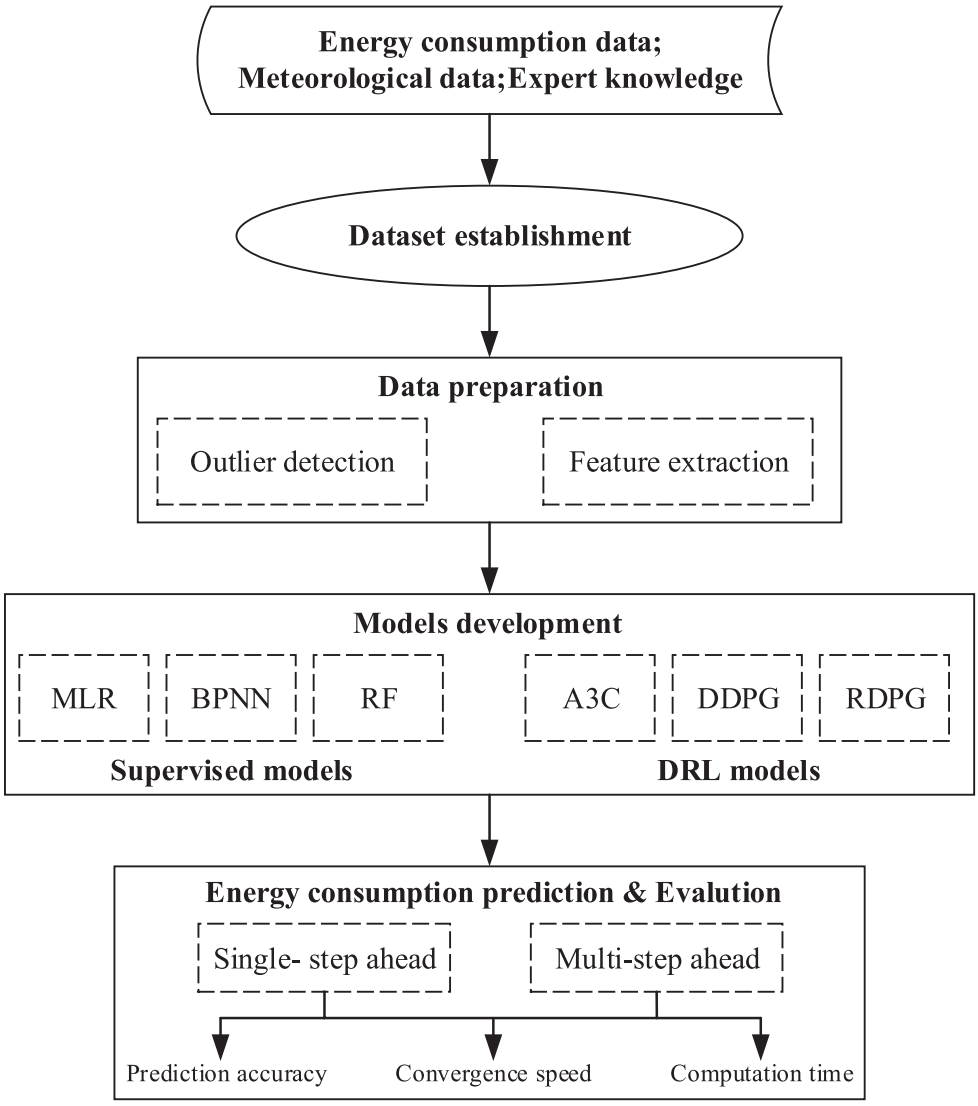

Fig. 1. Research outline.

framework [44]. Therefore one effective way to understand these three DRL techniques is to understand Actor-Critic.

Fig. 2 displays the schematic diagram of Actor-Critic. It can be observed that Actor-Critic contains two neural networks, namely Actor network and Critic network, respectively. Actor network with parameter θ is responsible to estimate the current policy (denoted as$\pi_\theta(s, a)$) and output an action based on its input state. Whereas the Critic network with parameter w is deployed to estimate the action-value function, which is then used to update the current policy (i.e. the parameters of Actor networks). In one training episode of Actor-Critic, agent interacts with environment based on current policy (i.e. Actor network) and gains transition $(s_t, a_t, s_{t+1}, a_{t+1}, r_{t+1})$. Then the action-value function of two tuples, i.e. (s_t, a_t) and (s_{t+1}, a_{t+1}), are computed by Critic network. Afterwards, TD-error (denoted as δ) can be calculated:

$$\delta = r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \tag{3}$$

It should be noted that if value function is used (e.g. A3C), the calculation of TD-error would follow as bellow:

$$\delta = r_{t+1} + \gamma V(s_{t+1}) - V(s_t) \tag{4}$$

Moreover, the parameters of Actor and Critic network can be updated using TD-error as follows:

$$Actor: d\theta = \alpha \nabla_\theta \log \pi_\theta(s, a) \cdot \delta \tag{5}$$

$$Critic: dw = \beta \cdot \frac{\partial \sum \delta^2}{\partial w} \tag{6}$$

where, $\nabla_\theta \log \pi_\theta(s, a)$ denotes score function, α and β represent the learning rates of Actor and Critic network, respectively.

2.3.3. A3C

Asynchronous Advantage Actor-Critic (A3C), which is an Actor-Critic based deep reinforcement learning framework, was proposed in 2016 [45]. A3C deploys a number of agents in parallel to calculate gradient simultaneously, each with their own environment. These agents perform asynchronous gradient descent to optimize the parameters of the same global agent, and each local agent periodically copies the parameters of global agent as their own parameters. Therefore, A3C is efficient and lightweight compared to other DRL techniques. A3C also overcomes the problem that traditional Actor-Critic is hard to converge due to the sample diversity caused by asynchronous learning.

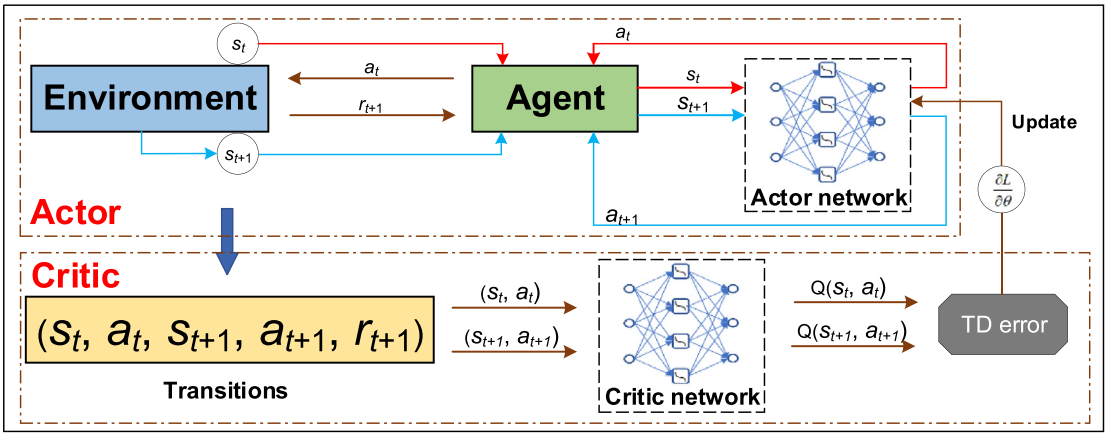

Fig. 2. Schematic diagram of Actor-Critic.

Table 1
Four neural networks in DDPG and their functions.

Neural network	Function
Actor network	Output a_t based on input s_t
Target Actor network	Output a_{t+1} based on input s_{t+1}
Critic network	Calculate action-value function $Q(s_t, a_t)$
Target Critic network	Calculate action-value function $Q(s_{t+1}, a_{t+1})$

2.3.4. DDPG

To improve the trainability of existing Actor-Critic based DRL algorithms, Lilllicrap et al. [46] proposed Deep Deterministic Policy Gradient (DDPG) method. Compared to Actor-Critic, there are three improvements in DDPG. Firstly, DDPG utilizes an experience buffer as memory device to store all historical transitions, and it learns in mini-batches randomly sampled from the experience buffer rather than learning online. Secondly, separate target networks for both Actor and Critic are established to weaken over estimation of action-value function. Consequently, there are four neural networks in DDPG. These four neural networks and their functions are listed in Table 1. Thirdly, the parameters of the two target networks are no longer directly copied from original networks, but updated by having them slowly track the original networks. This "soft" update strategy can constrain target networks from rapid change and make training process stable.

2.3.5. RDPG

In conventional DDPG method, multi-layer perceptron (MLP), which is consisted of multi-layer fully-connected networks, is deployed for both Actor network and Critic network. One limitation of that is the estimation of action-value function would not be accurate enough when coping with complicated control problem, due to the incompetence of MLP in capturing intricate nonlinearity. To ameliorate this problem, a novel method, namely RDPG, are proposed in this study. The only improvement of RDPG compared to DDPG is that RDPG uses Long Short-Term Memory (LSTM) which is an improved RNN to represent Critic and estimate action-value function. As shown in Fig. 3, LSTM stores information over long time periods by using purpose-built memory cell based on the structure of RNN. Three important "gates" (i.e. input gate, output gate, and forget gate) are designed to control the information flow inside each memory block. More details about LSTM can be seen in Ref. [47]. By using LSTM as Critic, RDPG can yield more accurate estimation of action-value function than DDPG method, as the Critic network for estimating $Q_w(s, a)$ is able to aggregate observations over time. Hence, more accurate TD-error can be provided for better update of Actor network.

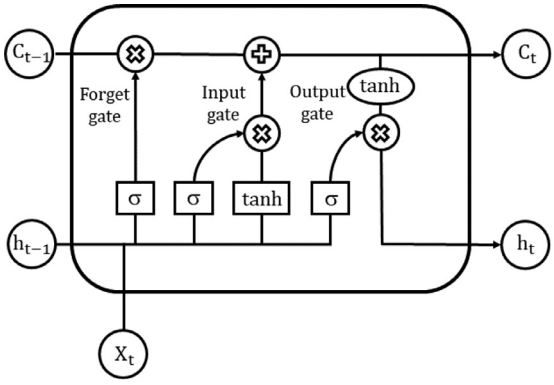

Fig. 3. Schematic diagram of LSTM.

2.4. Supervised prediction techniques

Three supervised prediction techniques, including multiple linear regression (MLR), Back Propagation Neural Network (BPNN) and Random Forest (RF), are also applied for building energy consumption forecasting. MLR is a linear technique while BPNN and RF are non-linear techniques. These three techniques are selected to compare prediction performance with above-mentioned DRL techniques due to their popularity in previous studies.

MLR attempts to capture the relationship between multiple arguments and one dependent variable in the form of: $y = a_0 + a_1x_1 + ... + a_nx_n$, thereby making the resulting model interpretable. In addition, MLR is regarded as an efficient technique and requires little computation load. The main drawback of MLR is the poor ability in coping with intricate nonlinearity. Hence, MLR serves as the performance benchmark in this study.

BPNN is one of the most commonly-used prediction technique, which has the same architecture as MLP and is trained by error back-propagation scheme. The error back-propagation is designed to minimize the mean square error between its actual outputs and expected outputs. It has been proven that BPNN is capable of solving linear and nonlinear problems with good accuracy and generalization.

With respect to RF, it's an ensemble prediction model which consists of a collection of Classification and Regression Trees (CART). These trees are independent with each other as their training data and input variables are randomly selected. The prediction result of RF is the mean of the predictions of its constituent trees. Using the ensemble of multiple trees with high diversity instead

of a single tree makes the model more stable and better prevent overfitting problem.

2.5. Evaluation indices

In this study, four evaluation indices are used to assess prediction accuracy of proposed models, including mean absolute error (MAE), root mean square error (RMSE), coefficient of determination (R^2) and coefficient of variance (CV). MAE describes the mean offset between actual values and predicted values by using absolute error, while RMSE denotes the standard deviation of the residuals of the actual and predicted values. Both MAE and RMSE are scale-dependent indices, and describe prediction errors in their original scale. By contrast, CV is scale-independent since the equation denominator is the mean of actual values, making it suitable for performance comparison with other studies. Moreover, R^2, which ranges from 0 to 1, measures the goodness of fit between actual values and predicted values. These four indices are formulated as bellow.

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - p_i| \tag{7}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - p_i)^2} \tag{8}$$

$$CV = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - p_i)^2}}{\bar{y}_i} \tag{9}$$

$$R^2 = \frac{\sum_{i=1}^{N}(p_i - \bar{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2} \tag{10}$$

where, $y_i$is the actual value at time point *i*, p_i is the predicted value at time point *i*, $\bar{y}_i$ denotes the mean of actual values, and *N* denotes the number of samples.

3. Case study

3.1. Case building description

The building studied in this paper is an office building situated in Henan province, China. It was put into use in 2014. The case building consists of nine floors and has a height of 42.9 m. In addition, it mainly comprises a variety of office rooms, equipment rooms and tool rooms, and its gross floor area is approximately 25000 square meters. The office hours approximately run from 08:00 to 17:30 every weekday (UTC+8).

Ground source heat pump (GSHP) system is deployed in the case building for refrigerating and heating. It should be mentioned that the energy consumption data used in this study was collected in summer, when the GSHP system was working in cooling mode. Accordingly, heating mode of the GSHP system is neglected in this study. Fig. 4 illustrates the schematic diagram of the GSHP system, which mainly consists submarine pumps, cyclone desanders, chillers, circulating pumps, water distributor, water collector as well as building terminals. The blue lines denote cooling water loop and the red lines denote water-supply pipes, whereas the green lines represent water-return pipes. When system works, groundwater is pumped by submarine pumps and cyclone desanders help filter the solid impurities in the water. Afterwards water is delivered to chillers. Chillers are the main device of the GSHP system as the heat exchange between groundwater and backwater from building terminals is accomplished here. After heat exchange, the groundwater is pumped to backwater wells, whereas the chilled water from chillers is sent to building terminals by water distributor. The circulating pumps play an important role in cycling supply and return water of building terminals.

3.2. Data description

The concerned building energy consumption (EC) data were collected from the GSHP system of case building, which comprises historic time series data from June 15 to July 27 in 2017 with a 5 min resolution. Note that the energy consumption in this paper refers to the sum of powers of all system devices, including chillers power, total pumps power (the sum of submarine pumps power and circulating pumps power), and auxiliary devices power. These powers values are measured by power meters embedded in the GSHP system. The energy consumption data contains 12384 observations in total. Besides, the local meteorological data are also collected from the local weather station, since outdoor weather conditions heavily influence building energy consumption as well. Hence, the introduction of meteorological data can enhance prediction accuracy and model robustness [21]. The collected meteorological data mainly consist of three variables: outdoor temperature (OT), wind speed (WS), and relative humidity (RH). The collected interval is 1 h.

In addition, an expert variable, namely system status (SS), is added according to the compressor power of the GSHP system. In other words, SS is a Boolean variable which equals to *True* when compressor is greater than zero and otherwise it equals to *False*. Considering that the Boolean values can't serve as model inputs, the SS variable is encoded into a one-hot representation. The reason for introducing this expert variable is that there is no complete control logic for the system operations and the on-off operations of the case GSHP system is controlled by building operation staffs. Therefore, expert variable is added, aiming to help better identify the switch status of the GSHP system operations.

Notably, there are other variables affecting the building energy consumption, which can be introduced as model inputs, such as building occupancy, global solar radiation, etc. The reason why this study doesn't use more input variables is twofold. Firstly, the input variables are sufficient, as the main focus of this study is to explore the predictive performance of DRL algorithms. Secondly, some of these variables are typically not available in practice. The summary of the main variables in this study are listed in Table 2.

3.3. Data preparation

Data preparation mainly contains two tasks, i.e. outlier detection and features extraction. Outliers in the energy consumption data should be removed prior to model development, since abnormal and low-quality data could exert negative effect on the final model performance. According to the compressor power, the raw energy consumption data are categorized into two types: system-on data and system-off data. LOF algorithm, which has been introduced in Section 2.2, is used for detecting the outliers from daily energy consumption profiles (i.e. system-on data). Fig. 5 presents partial results of outlier detection in the daily energy consumption profiles. A simple and feasible way to cope with the detected outliers is to replace them by linear interpolation. For system-off data, the energy consumption values are always a little larger than zero, which is caused by pre-heating logic in the compressor. Since the operation company decides to stop pre-heating to conserve energy, these non-zero data should be replaced by zero in system-off period [48].

Feature extraction is conducted by analyzing ACF and PACF of energy consumption time series data. Fig. 6 shows the ACF and

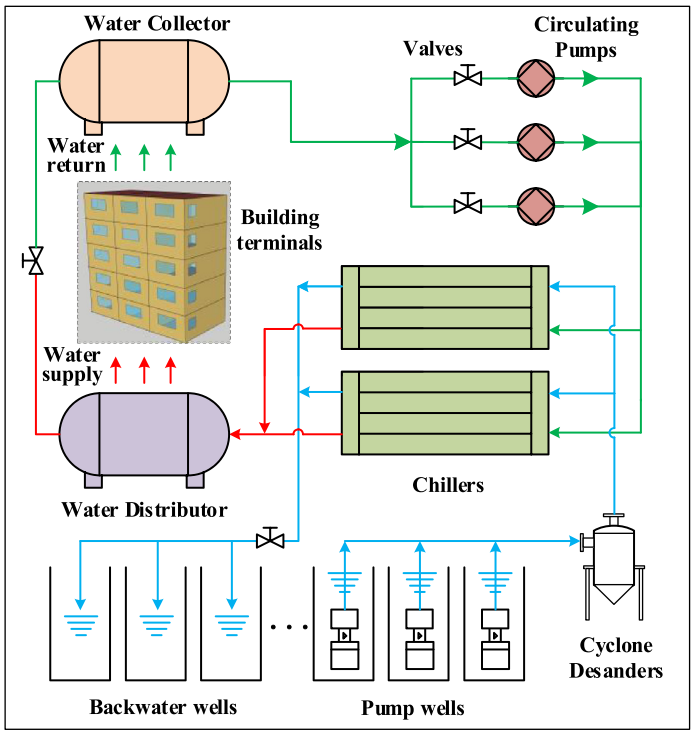

Fig. 4. Illustration of GSHP system.

Table 2
Summary of main variables.

Variables	Resolution	Data type	Units	Range
Energy consumption (EC)	5 min	numeric	kW	[0.00, 600.05]
Outdoor temperature (OT)	hourly	numeric	°C	[18.00, 38.00]
Wind speed (WS)	hourly	numeric	m/s	[0.00, 7.00]
Relative humidity (RH)	hourly	numeric	%	[24.63, 100.00]
System status (SS)	5 min	bool	/	{0, 1}

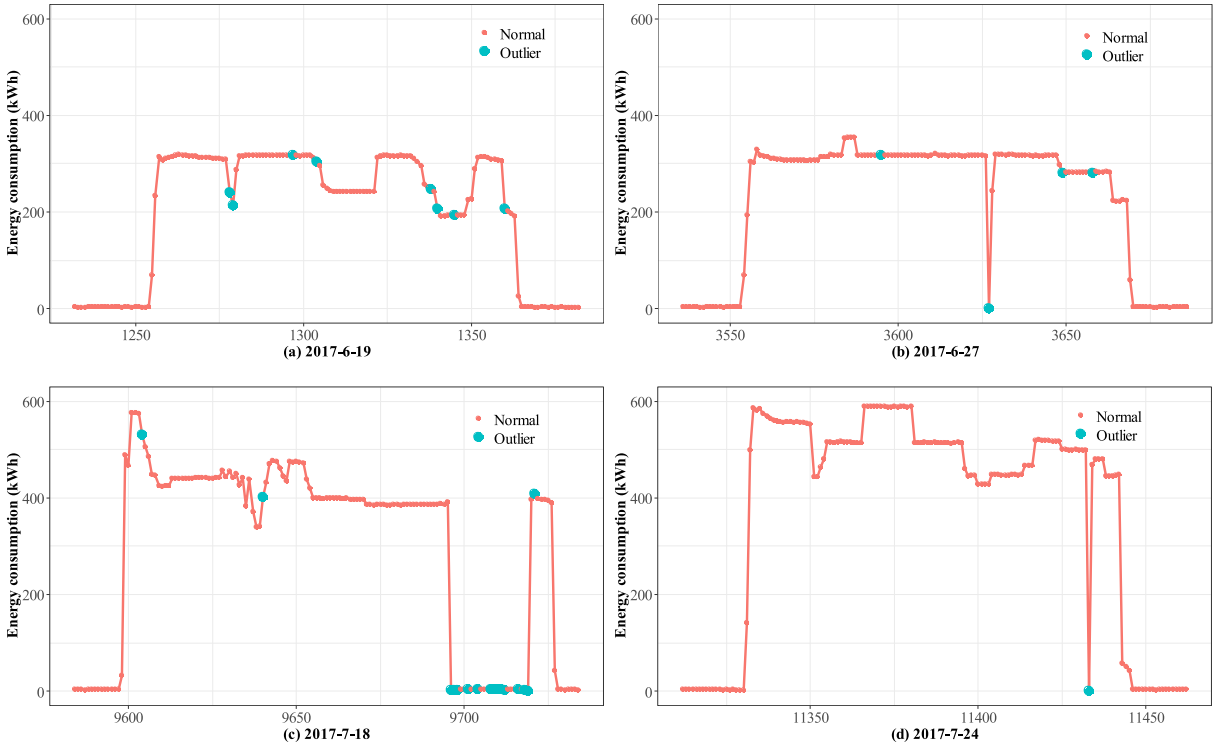

Fig. 5. Partial results of outlier detection using LOF method.

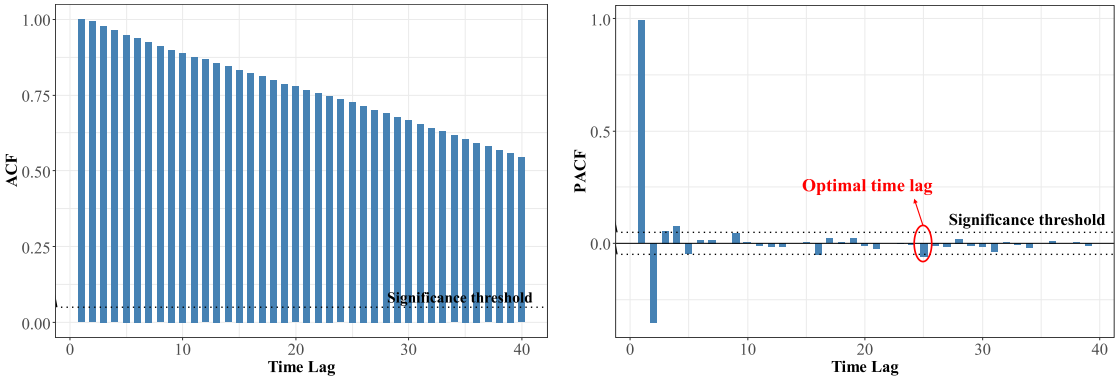

Fig. 6. ACF and PACF of energy consumption.

PACF diagrams of building energy consumption. The black dotted lines denote the significance threshold, which is set to 5% in this study. Each time step exceeding that threshold is considered heavily related to the current energy consumption. The maximum time lag is set to 40. According to the ACF and PACF results, it can be found that the PACF of building energy consumption is no longer significant when the time lag is greater than 25. Therefore, the optimal time lag period is chosen to be 25. It should be mentioned that only the first 5 time steps are chosen to serve as model inputs in the latter single-step ahead prediction, because the single-step ahead prediction is a simple task and 5 time steps are sufficient. As opposed to that, all 25 time steps are selected as the model inputs for multi-step ahead prediction. The chosen 5 time steps for single-step ahead prediction or 25 time steps for multi-step ahead prediction are all historical data points of EC variable, which is one of the main variables in this study (as shown in Table 2). Apart from EC variable, the other four variables in Table 2 (i.e. OT, WS, RH and SS) are also introduced into model inputs, to enhance model accuracy and robustness. Only the data point at current time step of these four variables are selected as input features. To sum up, single-step ahead forecasting has 9 input features (i.e. five historical data points of EC variable, three data points of three meteorological variables and one data point of the expert variable at current time step), whereas multi-step ahead forecasting has 29 input features (i.e. 25 historical data points of EC variable, three data points of three meteorological variables and one data point of the expert variable at current time step).

In addition, local meteorological data is completed using linear interpolation method, in order to make meteorological data have the same observation number as other variables in the dataset. Data transformation is another important task in data preparation process. Data normalization is the main transformation type and the main purpose is to make each input feature in the similar scale, helping to find the global optimum by Stochastic Gradient Descent when prediction techniques are applied. Fig. 7 displays the hourly heat map of the building energy consumption data after data normalization for 43 days. The horizontal axis denotes the 43 days while the vertical axis denotes the 24 h in each day. The blocks in the picture represent the hourly energy consumption of the case building. More notably, the blocks in red, white and blue respectively indicate that the associated energy consumption is relatively high, medium and low.

Finally, the dataset is partitioned into two parts with a ratio of 0.75:0.25 for models training and testing purposes.

3.4. Development and optimization of prediction models

3.4.1. Development of prediction models

In this study, all models are developed for single-step ahead forecasting (5 min in advance) and multi-step ahead forecasting (1 h in advance). As for three supervised models, their establishment process is similar to previous studies. The model inputs are the extracted features set at each time step while the output is the corresponding building energy consumption.

The development of DRL prediction models is one of the key parts in this study. To apply DRL for building energy consumption forecasting, energy consumption prediction problem should be translated into a RL control problem. Therefore, the meaning of state, action and reward should be defined. The state at each time step is represented by the extracted features set as the input of supervised models. Take single-step ahead prediction as an example, the state at time T is represented by vector $[EC_{T-4}, EC_{T-3}, EC_{T-2}, EC_{T-1}, EC_T, OT_T, WS_T, RH_T, SS_T]$. The state space consists of the state of each time step. And the action space consists of the continuous energy consumption values ranging from 0 to 610 (this range is set according to historic data). During training process, artificial agent outputs an energy consumption value from [0,610] based on its observed state. Note that the output energy consumption value is exactly the predicted value (single-step ahead or multi-step ahead). A method to inform agent whether its prediction is accurate is to set a reasonable reward function. In this study, the reward function is set as below.

$$r_{t+1} = -|EC_t - a_t| \tag{11}$$

where, the EC_t denotes the real energy consumption value at time step t, a_t denotes the action performed by agent (i.e. the predicted energy consumption value) at time step t. If the output action is close to real energy consumption, the reward would be close to zero, and otherwise the reward would be far away from zero in negative direction.

Once the energy consumption prediction problem is transformed into decision-making problem, DRL techniques can be applied to solve it. The main training processes of A3C, DDPG and RDPG are respectively presented in Tables 3 and 4. The training process of RDPG is very similar to that of DDPG. The only difference is that RDPG adopts LSTM for the development of Critic network. Further details of the development of DRL models can be seen in Ref. [49].

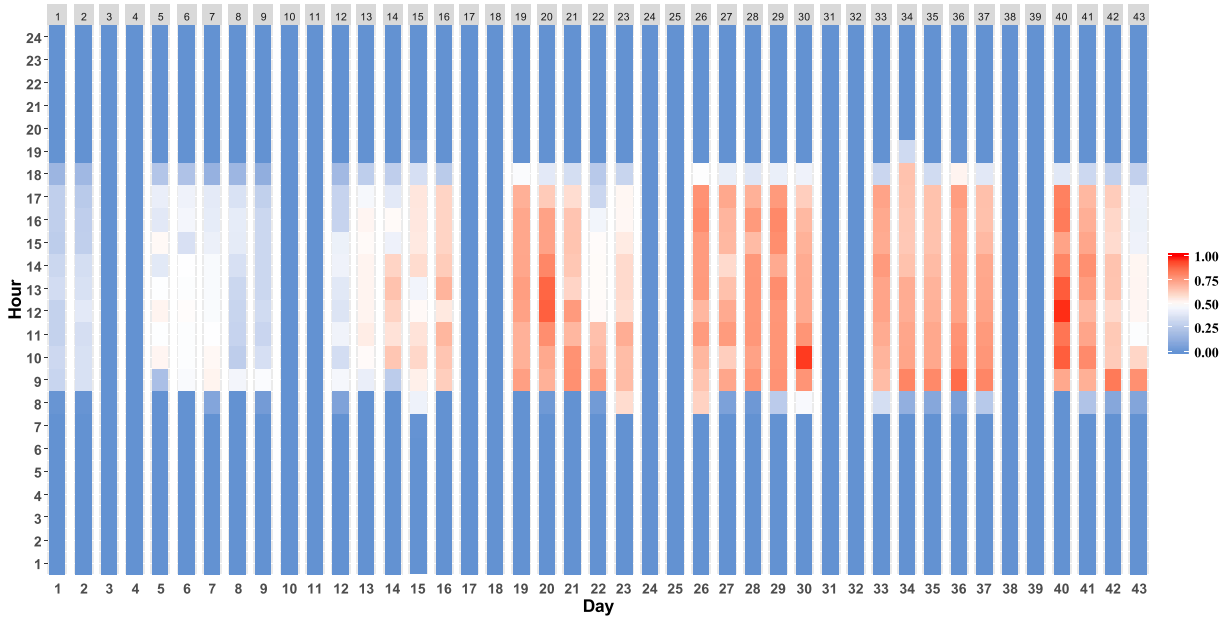

Fig. 7. The hourly heat map of the building energy consumption data for 43 days.

Table 3
Pseudo code of A3C for energy consumption forecasting.

Pseudo code: A3C
Initialize the global Actor and Critic networks with parameters θ and w
Initialize the thread-specific networks with parameters θ' and w'
Repeat (for each episode)
Reset gradient: $d\theta \leftarrow 0$; $dw \leftarrow 0$
Synchronize thread-specific parameters: $\theta' \leftarrow \theta$; $w' \leftarrow w$
Randomly select an initial state s (denoted as s_t) from state space
Repeat (for each step)
Choose a_t from s_t according to Actor network (Note: a_t is the predicted value)
Execute action a_t, receive immediate reward r_t according to Eq. (9)
Set the subsequent state to s_{t+1}
Until terminal s_t **or** maximum number of steps is reached
Compute the value function of the last state using thread-specific Critic:
$V(s_t) = \begin{cases} 0 & \text{for terminal } s_t \\ V_{w'}(s_t) & \text{for non-terminal } s_t \end{cases}$
for each step (from last state to initial state)
compute TD-error of each step using Eq. (4)
accumulate Actor's local gradient using Eq. (5)
accumulate Critic's local gradient using Eq. (6)
end for
Perform asynchronous update of θ and w using local accumulative gradient
Until maximum number of episodes is reached

3.4.2. Optimization of model parameters

Model parameters can greatly influence the model performance to some extent. For three supervised models, parameters of MLR are determined by using the least squares principle and there are no parameters required to be optimized. BPNN has two parameters required optimization, i.e. the size of hidden neurons and the activation function of each layer. Its number of layers is set to 3. In addition, two parameters, i.e. the maximum tree depth and the maximum number of features to consider when looking for the best splitting, should be optimized for RF model. And to guarantee the performance of RF model, the number of base trees is set to 1000.

Compared to supervised models, DRL models have more complex training schemes and more parameters required to be optimized. The optimizations of Actor and Critic networks structures are the pivotal ingredient in Actor-Critic based DRL models optimization process. For the three DRL models studied in this paper, the number of layers of both Actor and Critic is fixed as 3 to make a fair comparison with supervised models. And other four important parameters (i.e. the sizes of hidden neurons and activation functions of each layer in both Actor and Critic) are considered. In this study, all the parameters optimization processes are performed based on genetic algorithm. After extensive numerical experiments, the optimization results of the six models are summarized in Table 5.

4. Results and discussion

In this section, the prediction performances of the six techniques, including three DRL models and three supervised models are presented and compared. The forecast time horizon is first set to 5 min to explore the single-step ahead prediction performance of these six models, then 1 h is chosen to verify their multi-step ahead prediction performance. The comparative analysis is conducted from three perspectives, i.e. prediction accuracy, conver-

Table 4
Pseudo code of DDPG/RDPG for energy consumption forecasting.

Pseudo code: DDPG/RDPG
Initialize Actor and Critic networks with parameters θ and w
Initialize target Actor and Critic networks with parameters: $\theta' \leftarrow \theta$, $w' \leftarrow w$
Repeat (for each episode)
Randomly select an initial state s (denoted as s_t) from state space
Repeat (for each step)
Choose a_t from s_t according to Actor network (Note: a_t is the predicted value)
Execute action a_t, receive immediate reward r_t according to Eq. (9)
Set the subsequent state to s_{t+1}
Choose a_{t+1} from s_{t+1} according to target Actor network
Store transition $(s_t, a_t, s_{t+1}, a_{t+1}, r_t)$ in experience buffer
Sample a mini-batch from experience buffer
Compute action-value functions $Q(s_t, a_t)$ using Critic network
Compute action-value function $Q(s_{t+1}, a_{t+1})$ using target Critic network
Compute TD-error using Eq. (3)
Update Actor network using Eq. (5)
Update Critic network using Eq. (6)
Update target networks (soft update)
$s_t \leftarrow s_{t+1}$
Until terminal s_t **or** maximum number of steps is reached
Until maximum number of episodes is reached

Table 5
Parameters optimization results of the six models

Model	Parameter	Results (single-step ahead)	Results (multi-step ahead)
MLR	/	/	/
BPNN	Neurons	9, 23, 1	18, 46, 1
	Activation function	*Relu* / *Linear*	*Relu* / *Linear*
RF	Tree depth	5	6
	Maximum features	8	13
A3C	Neurons (Actor)	9, 52, 2	18, 75, 2
	Activation function (Actor)	*Relu* / *Tanh*	*Relu* / *Tanh*
	Neurons (Critic)	10, 48, 1	19, 79, 1
	Activation function (Critic)	*Relu* / *Linear*	*Relu* / *Linear*
DDPG	Neurons (Actor)	9, 46, 1	18, 64, 1
	Activation function (Actor)	*Relu* / *Sigmoid*	*Relu* / *Sigmoid*
	Neurons (Critic)	10, 42, 1	19, 74, 1
	Activation function (Critic)	*Relu* / *Linear*	*Relu* / *Linear*
RDPG	Neurons (Actor)	9, 54, 1	18, 86, 1
	Activation function (Actor)	*Relu* / *Sigmoid*	*Relu* / *Sigmoid*
	Neurons (Critic)	10, 58, 1	19, 80, 1
	Activation function (Critic)	*Relu* / *Linear*	*Relu* / *Linear*

gence speed as well as computation time. The detailed results and discussion are exhibited below.

4.1. Single-step ahead prediction

In single-step ahead prediction, the forecast time horizon is chosen to be 5 min, which can inform facility operation staffs of the real-time information of energy consumption fluctuation. Fig. 8 displays the prediction results of the six models in single-step ahead forecasting. The solid lines denote ideal fitting lines, which indicate the predicted value is equal to measured value. The two dotted lines represent ±20% error lines, which indicate the predicted value is 20% larger or 20% smaller than measured value. It can be observed that BPNN and RF obviously outperform the benchmark model (i.e. MLR), which indicates that non-linear techniques can yield more promising results than linear techniques. In regard to three DRL models, A3C perform worst among these six models. The prediction result of A3C is even worse than MLR. By contrast, the other two DRL models, i.e. DDPG and RDPG, predict the building energy consumption 5 min in advance with flying colors. For better comparison of these six models, the four evaluation indices described in Section 2.5 are used to assess their prediction accuracy on testing data and the results are listed in Table 6. The evaluation results of the predictions show that the prediction performance of A3C measured by MAE, RMSE, R^2 and CV are 73.18%, 118.76%, 6.00% and 118.75% worse than MLR, respectively, indicating the incompetence of A3C in single-step ahead forecasting. DDPG and RDPG models outperform the three supervised models and have the best performances in all indices. The result that RDPG model doesn't show advantages over DDPG is not in correspondence with expectation. A possible explanation is that the single-step ahead prediction task is simple and the MLP network in DDPG is sufficient to capture the nonlinearity.

Table 6
Single-step ahead prediction accuracy on testing data

Model	MAE	RMSE	R^2	CV
MLR	8.527	25.481	0.984	17.526%
BPNN	5.465	17.244	0.992	11.860%
RF	5.625	17.885	0.992	12.301%
A3C	14.767	55.742	0.925	38.339%
DDPG	4.550	17.390	0.993	11.961%
RDPG	4.700	16.668	0.993	11.464%

Fig. 9 presents the prediction error varying tendencies of three DRL models in single-step ahead forecasting. It can be intuitive to see that the prediction errors of DDPG and RDPG decrease rapidly near the 100th iteration, and both of these two models have converged before the 200th iteration. DDPG converges fastest among these three models and RDPG converges slightly slower

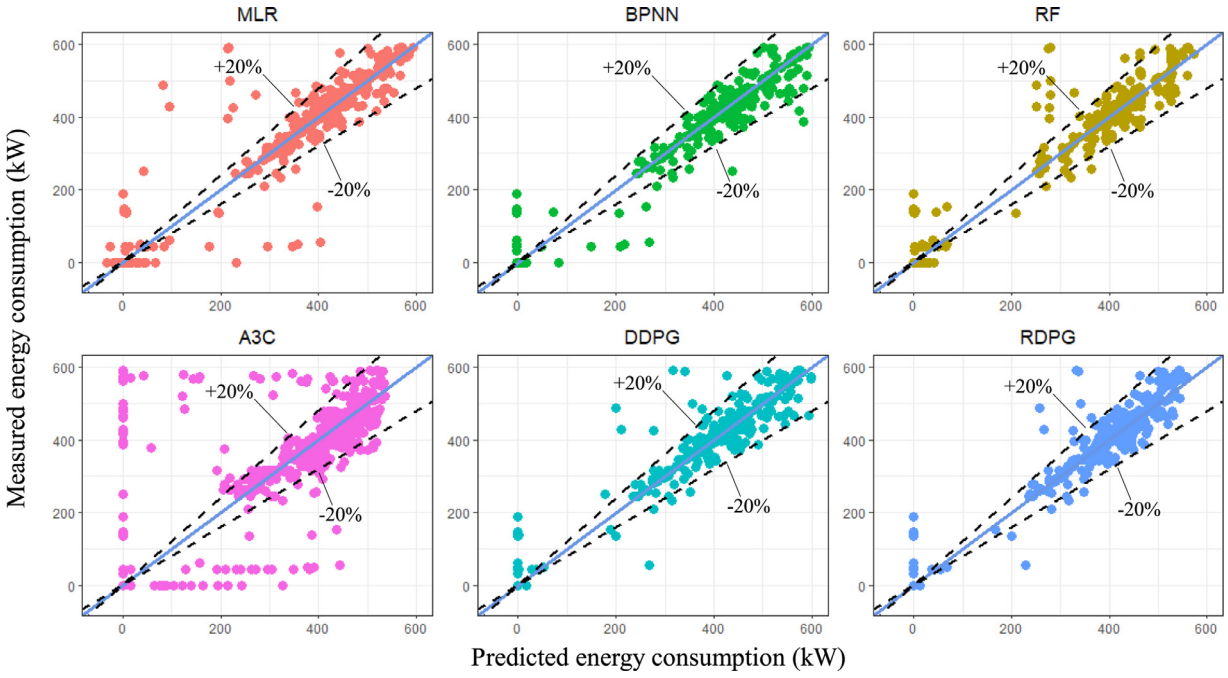

Fig. 8. Single-step ahead prediction results of the six models.

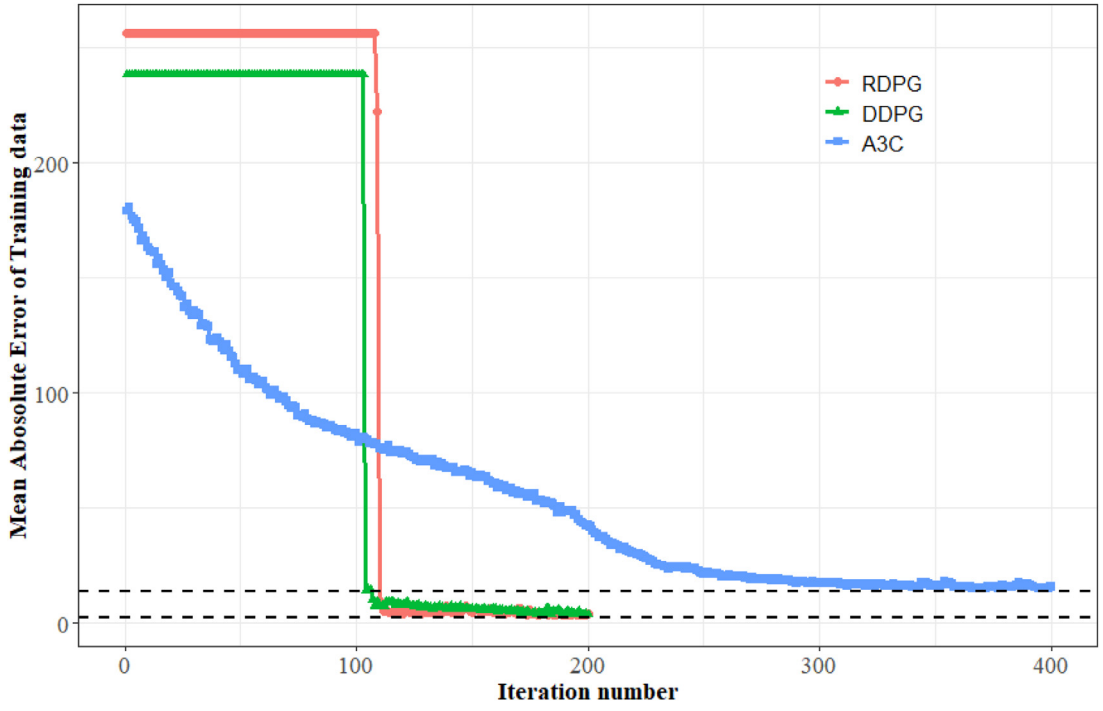

Fig. 9. Prediction error varying tendencies of three DRL models in single-step ahead forecasting.

than DDPG. By contrast, the prediction error of A3C decreases much slower than the other two models and converges at nearly 400th iteration. Therefore, the maximum number of iterations is set to 200 for DDPG and RDPG, and 400 for A3C.

Moreover, the computation times of these six models are also investigated. The computation times spent for model training and energy consumption predicting are listed in Table 7. The processor for computation in this study is 3.20 GHz Intel Core i5-4570. And all the computation is conducted in Python 3.7 using the neural network construction package *Tensorflow* [50]. As can be seen from

Table 7
Computation time of these six techniques in single-step ahead forecasting

Model	Time for training	Time for prediction
MLR	0.003	0.001
BPNN	27.559	0.033
RF	9.361	0.095
A3C	76.576	0.639
DDPG	161.875	0.534
RDPG	1594.006	0.523

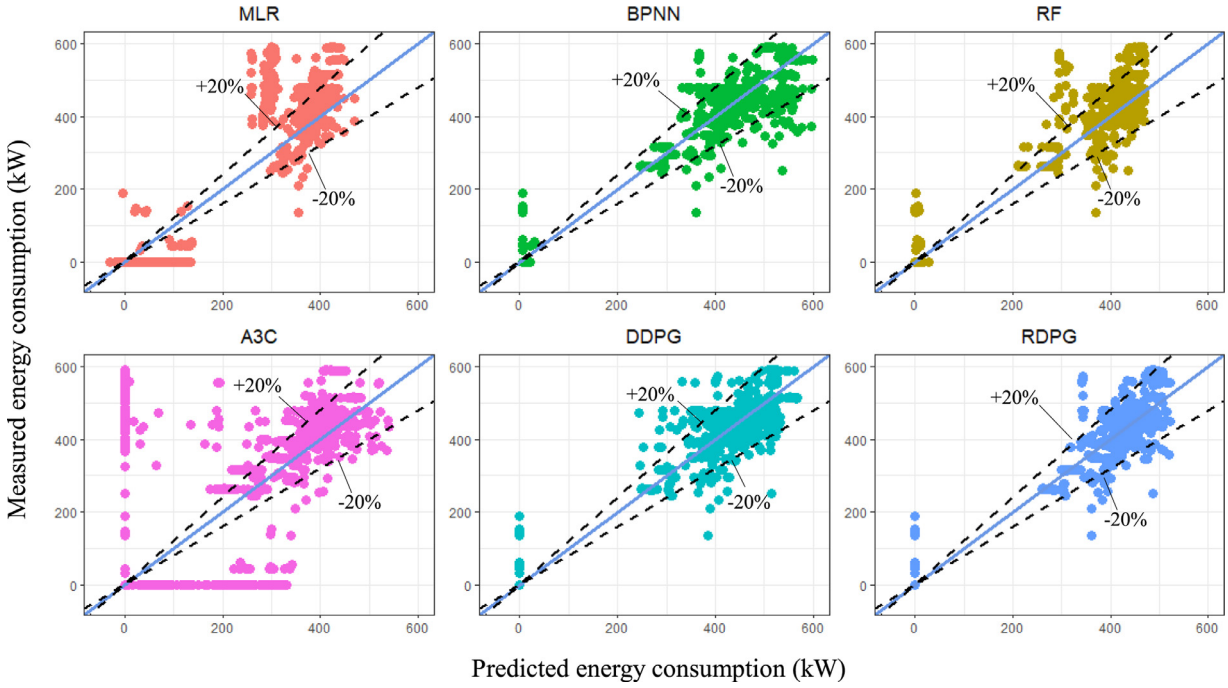

Fig. 10. Multi-step ahead prediction results of the six models.

Table 8
Multi-step ahead prediction accuracy on testing data

Model	MAE	RMSE	R^2	CV
MLR	38.806	57.574	0.920	39.599%
BPNN	15.745	31.165	0.977	21.435%
RF	16.012	34.834	0.971	23.959%
A3C	42.430	112.104	0.698	77.105%
DDPG	13.175	30.440	0.978	20.937%
RDPG	12.218	28.787	0.980	19.800%

Table 9
Computation time of these six techniques in multi-step ahead forecasting

Model	Time for training	Time for prediction
MLR	0.006	0.001
BPNN	30.991	0.050
RF	13.336	0.107
A3C	81.586	0.643
DDPG	177.815	0.530
RDPG	1618.567	0.540

Table 7, both the training times and predicting times of DRL models are much larger than those of supervised models. Linear technique (i.e. MLR) needs less computation time than other non-linear techniques. RDPG is the most computational expensive one among these six models due to the complex training scheme of LSTM. Besides, A3C is the most efficient one among three DRL models, even though its training process needs more iterations. Once the models have been established, the computation time for predicting is so short compared with time spent for training that it can be neglected.

4.2. Multi-step ahead prediction

In regard to multi-step ahead forecasting, the building energy consumption is forecasted 1 hour in advance. Fig. 10 exhibits the multi-step ahead prediction results of these six models. It can be found the performances of multi-step ahead prediction is worse than those of single-step ahead prediction. The reason for this is that the dependency of a time point in a time series variable on its past becomes weaker as the forecast time horizon increases. MLR doesn't show a reasonable accuracy compared to BPNN and RF as it does in single-step ahead forecasting task, since the nonlinearity in multi-step ahead prediction is hard to capture using linear technique. With respect to DRL models, A3C still performs a poor prediction result, whereas RDPG shows an evident advantage over other five models. The resulting MAE, RMSE, R^2 and CV are summarized in Table 8. BPNN is the most accurate model among three supervised models. More notably, compared to BPNN, RDPG model can enhance the prediction accuracy and the resulting CV can be below 20%. RDPG performs better than the three supervised models with MAE, RMSE, R^2 and CV of at least 22.40%, 7.63%, 0.31% and 7.63%, respectively. The results indicate that the introduction of LSTM can greatly boost the prediction accuracy in multi-step ahead prediction.

Fig. 11 exhibits the prediction error varying tendencies of three DRL models in multi-step ahead forecasting. Similar but different from the above single-step ahead predictions, DDPG as well as RDPG shows fast convergence speed, while A3C shows a much slower convergence speed than DDPG and RDPG. It should be noted that the final convergent error of RDPG is obviously lower than those of DDPG and A3C, indicating a better training performance of RDPG.

Computation times of these six techniques in multi-step ahead forecasting are listed in Table 9. Compared to single-step ahead prediction, multi-step ahead prediction costs more computation time in models training, which is attributed to the increase of input features and consequent increase of model parameters required to be optimize. Similar findings can be obtained as above-mentioned single-step ahead prediction. For instance, A3C is the most efficient technique among the three DRL models, and RDPG is the most time-consuming one.

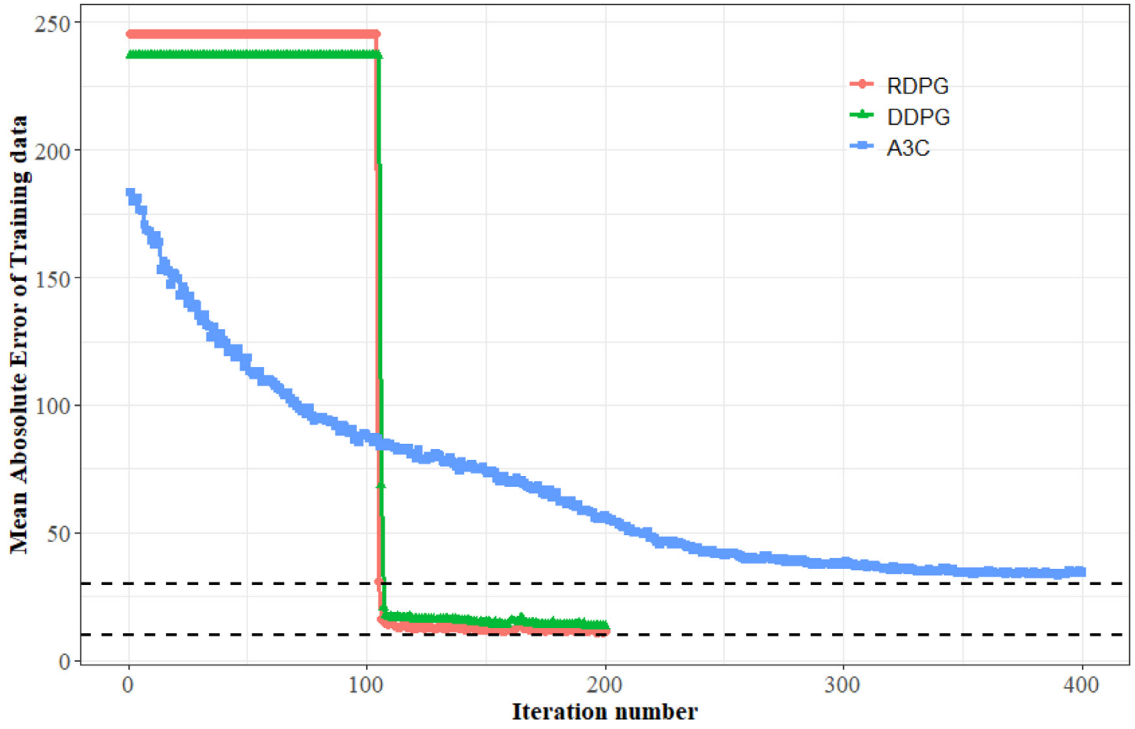

Fig. 11. Prediction error varying tendencies of three DRL models in multi-step ahead forecasting.

5. Conclusion

This paper comprehensively investigates the potential of DRL techniques in forecasting building energy consumption. Three commonly-used DRL techniques (i.e. A3C, DDPG and RDPG) are applied for both single-step ahead forecasting and multi-step ahead forecasting. Comparing with three common supervised models (i.e. MLR, BPNN and RF), a rounded analysis between these six models is conducted from three perspectives, i.e. prediction accuracy, convergence speed as well as computation time.

The research results show that DDPG outperforms supervised models both in single-step ahead prediction and multi-step ahead prediction. RDPG model doesn't have advantages over DDPG in single-step ahead prediction, yet leads to evident accuracy improvement in multi-step ahead prediction. A possible reason is that the nonlinear relationship in single-step ahead prediction is simple and DDPG is sufficient to capture. By contrast, A3C presents poor performances both in single-step ahead prediction and multi-step ahead prediction, indicating its incompetence in forecasting building energy consumption. Among these three DRL techniques, both DDPG and RDPG converge quickly while A3C shows a much slower convergence speed. In terms of computation time, DRL models account for more computation time for model training compared with supervised models due to their more complex training schemes. Among three DRL models, RDPG takes the most computational time for model training while A3C is the most efficient one, although A3C requires more training iterations.

This work demonstrates that DRL techniques have great potential in the application for building energy consumption prediction. Further studies will focus on exploring the prediction performances of DRL methods in medium and long-term building energy consumption prediction.

Declaration of Competing Interest

The authors declared that they have no conflicts of interest to this work.

Acknowledgments

The authors gratefully acknowledge the support of National Natural Science Foundation of China (Grant Nos. 51876070 and 51576074), and State Key Laboratory of Air-Conditioning Equipment and System Energy Conservation (SKLACKF201606).

References

[1] X. Cao, X. Dai, J. Liu, Building energy-consumption status worldwide and the state-of-the-art technologies for zero-energy buildings during the past decade, Energy Build. 128 (2016) 198–213 /09/15/ 2016.
[2] T. Huo, et al., China's energy consumption in the building sector: a statistical yearbook-energy balance sheet based splitting method, J. Clean. Prod. 185 (2018) 665–679 /06/01/ 2018.
[3] B. Becerik-Gerber, et al., Civil engineering grand challenges: opportunities for data sensing, Inf. Anal. Knowl. Discov. 28 (4) (2013) 04014013.
[4] D.c. Gao, S. Wang, K. Shan, C. Yan, A system-level fault detection and diagnosis method for low delta-T syndrome in the complex HVAC systems, Appl. Energy 164 (2016) 1028–1038 /02/15/ 2016.
[5] K. Shan, S. Wang, D.c. Gao, F. Xiao, Development and validation of an effective and robust chiller sequence control strategy using data-driven models, Autom. Constr. 65 (2016) 78–85 /05/01/ 2016.
[6] X. Xue, S. Wang, Y. Sun, F. Xiao, An interactive building power demand management strategy for facilitating smart grid optimization, Appl. Energy 116 (2014) 297–310 /03/01/ 2014.
[7] A. Colmenar-Santos, L.N. Terán de Lober, D. Borge-Diez, M. Castro-Gil, Solutions to reduce energy consumption in the management of large buildings, Energy Build. 56 (2013) 66–77 /01/01/ 2013.
[8] K. Amasyali, N.M. El-Gohary, A review of data-driven building energy consumption prediction studies, Renew. Sustain. Energy Rev. 81 (2018) 1192–1205 /01/01/ 2018.
[9] Z. Wang, Y. Wang, R.S. Srinivasan, A novel ensemble learning approach to support building energy use prediction, Energy Build. 159 (2018) 109–122 /01/15/ 2018.
[10] A.S. Ahmad, et al., A review on applications of ANN and SVM for building electrical energy consumption forecasting, Renew. Sustain. Energy Rev. 33 (2014) 102–109 /05/01/ 2014.
[11] C. Fan, F. Xiao, Y. Zhao, A short-term building cooling load prediction method using deep learning algorithms, Appl. Energy 195 (2017) 222–233 /06/01/ 2017.
[12] C. Fan, F. Xiao, S. Wang, Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques, %J. Appl. Energy 127 (2014).
[13] M. Brown, C. Barrington-Leigh, Z.J.J. o. B. P. S. Brown, Kernel regression for real-time building, Energy Anal. 5 (4) (2012) 263–276.

[14] J. Liu, J. Wang, G. Li, H. Chen, L. Shen, L.J. Xing, Evaluation of the energy performance of variable refrigerant flow systems using dynamic energy benchmarks based on data mining techniques, Appl Energ. 208 (2017) 522–539.
[15] M.M. Breunig, H.P. Kriegel, R.T. Ng, J. Sander, LOF: identifying density-based local outliers, ACM Sigmod Record 29 (2) (2000) 93–104 ACM.
[16] Y. Chen and W.J.N.R.R. Wu, "Isolation forest as an alternative data-driven mineral prospectivity mapping method with a higher data-processing efficiency," vol. 28, no. 1, pp. 31–46, 2019.
[17] K. Yan, Z. Ji, W. Shen, Online fault detection methods for chillers combining extended kalman filter and recursive one-class SVM, Neurocomputing 228 (2017) 205–212 /03/08/ 2017.
[18] K. Grolinger, A. L'Heureux, M.A.M. Capretz, L. Seewald, Energy forecasting for event venues: big data and prediction accuracy, Energy Build. 112 (2016) 222–233 /01/15/ 2016.
[19] C. Lemke, B. Gabrys, Meta-learning for time series forecasting and forecast combination, Neurocomputing 73 (10) (2010) 2006–2016 /06/01/ 2010.
[20] Y. Wang, et al., Load profiling and its application to demand response: A review, Tsinghua Sci Technol. 20 (2) (2015) 117–129.
[21] C. Fan, Y. Sun, Y. Zhao, M. Song, J. Wang, Deep learning-based feature engineering methods for improved building energy prediction, Appl. Energy 240 (2019) 35–45 /04/15/ 2019.
[22] C. Tian, C. Li, G. Zhang, Y. Lv, Data driven parallel prediction of building energy consumption using generative adversarial nets, Energy Build. 186 (2019) 230–243 /03/01/ 2019.
[23] A.H. Neto, F.A.S. Fiorelli, Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption, Energy Build. 40 (12) (2008) 2169–2176 /01/01/ 2008.
[24] Q. Li, Q. Meng, J. Cai, H. Yoshino, A. Mochida, Applying support vector machine to predict hourly cooling load in the building, Appl. Energy 86 (10) (2009) 2249–2256 /10/01/ 2009.
[25] Z. Yu, F. Haghighat, B.C.M. Fung, H. Yoshino, A decision tree method for building energy demand modeling, Energy Build. 42 (10) (2010) 1637–1646 /10/01/ 2010.
[26] C. Xu, et al., Modal decomposition based ensemble learning for ground source heat pump systems load forecasting, Energy Build. 194 (2019) 62–74 /07/01/ 2019.
[27] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436 05/27/online.
[28] R.F. Berriel, A.T. Lopes, A. Rodrigues, F.M. Varejão, T. Oliveira-Santos, Monthly energy consumption forecast: a deep learning approach, in: International Joint Conference on Neural Networks (IJCNN), 2017, 2017, pp. 4283–4290.
[29] G. Fu, Deep belief network based ensemble approach for cooling load forecasting of air-conditioning system, Energy 148 (2018) 269–282 /04/01/ 2018.
[30] A. Rahman, V. Srikumar, A.D. Smith, Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks, Appl. Energy 212 (2018) 372–385 /02/15/ 2018.
[31] E. Mocanu, P.H. Nguyen, M. Gibescu, W.L. Kling, Deep learning for estimating building energy consumption, Sustain. Energy Grids Netw. 6 (2016) 91–99 /06/01/ 2016.
[32] V. Mnih et al., "Playing Atari with Deep Reinforcement Learning," 2013.
[33] M. Asada, et al., Purposive behavior acquisition for a real robot by vision-based reinforcement learning, Mach Learn. 23 (2-3) (1996) 279–303.
[34] M. Kuderer, S. Gulati, W. Burgard, Learning driving styles for autonomous vehicles from demonstration, in: IEEE International Conference on Robotics and Automation (ICRA), 2015, IEEE, 2015, pp. 2641–2646.
[35] K. Dalamagkidis, D. Kolokotsa, K. Kalaitzakis, G.S. Stavrakakis, Reinforcement learning for energy conservation and comfort in buildings, Build. Environ. 42 (7) (2007) 2686–2698 /07/01/ 2007.
[36] A. Nagy, H. Kazmi, F. Cheaib, and J.J. a. p.a. Driesen, "Deep Reinforcement Learning for Optimal Control of Space Heating," 2018.
[37] L. Yang, Z. Nagy, P. Goffin, A. Schlueter, Reinforcement learning for optimal control of low exergy buildings, Appl. Energy 156 (2015) 577–586 10/15/ 2015.
[38] A. Diez-Olivan, J.A. Pagan, R. Sanz, B. Sierra, Data-driven prognostics using a combination of constrained K-means clustering, fuzzy modeling and LOF-based score, Neurocomputing 241 (2017) 97–107 /06/07/ 2017.
[39] S. Sun, et al., Optimization of support vector regression model based on outlier detection methods for predicting electricity consumption of a public building WSHP system, Energy Build. 151 (2017) 35–44 /09/15/ 2017.
[40] L.P. Kaelbling, M.L. Littman, and A.W.J.J. o. a. i. r. Moore, "Reinforcement learning: a survey," vol. 4, pp. 237–285, 1996.
[41] M. Babaeizadeh, I. Frosio, S. Tyree, J. Clemons, and J.J.C. a. Kautz, "GA3C: GPU-Based A3C for Deep Reinforcement Learning," 2016.
[42] N. Heess, J.J. Hunt, T.P. Lillicrap, and D.J. a. p. a. Silver, "Memory-Based Control With Recurrent Neural Networks," 2015.
[43] L.P. Tuyen, T. Chung, Controlling bicycle using deep deterministic policy gradient algorithm, in: 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), 2017, 2017, pp. 413–417.
[44] V.R. Konda, J.N. Tsitsiklis, Actor-critic algorithms, in: Advances in Neural Information Processing Systems, 2000, pp. 1008–1014.
[45] V. Mnih, et al., Asynchronous methods for deep reinforcement learning, in: International Conference on Machine Learning, 2016, pp. 1928–1937.
[46] T.P. Lillicrap et al., "Continuous Control with Deep Reinforcement Learning," 2015.
[47] S. Hochreiter and J.J.N. c. Schmidhuber, "Long short-term memory," vol. 9, no. 8, pp. 1735–1780, 1997.
[48] J. Wang, et al., Energy consumption prediction for water-source heat pump system using pattern recognition-based algorithms, Appl. Therm. Eng. 136 (2018) 755–766 /05/25/ 2018.
[49] T. Liu, C. Xu, Y. Guo, H. Chen, A novel deep reinforcement learning based methodology for short-term HVAC system energy consumption prediction, Int. J. Refrig. (2019) 2019/07/27/.
[50] M. Abadi, et al., Tensorflow: a system for large-scale machine learning, in: 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16), 2016, pp. 265–283.