



OPEN ACCESS

EDITED BY
Yikui Liu,
Stevens Institute of Technology,
United States

REVIEWED BY
Zhengmao Li,
Nanyang Technological University,
Singapore
Zao Tang,
Hangzhou Dianzi University, China

*CORRESPONDENCE
Fujia Han,
hanfujia@epri.sgcc.com.cn

SPECIALTY SECTION
This article was submitted to Smart
Grids,
a section of the journal
Frontiers in Energy Research

RECEIVED 05 July 2022
ACCEPTED 25 July 2022
PUBLISHED 05 January 2023

CITATION
Han F and Wang X (2023), Adaptive
individual residential load forecasting
based on deep learning and dynamic
mirror descent.
Front. Energy Res. 10:986146.
doi: 10.3389/fenrg.2022.986146

COPYRIGHT
© 2023 Han and Wang. This is an open-
access article distributed under the
terms of the [Creative Commons
Attribution License \(CC BY\)](#). The use,
distribution or reproduction in other
forums is permitted, provided the
original author(s) and the copyright
owner(s) are credited and that the
original publication in this journal is
cited, in accordance with accepted
academic practice. No use, distribution
or reproduction is permitted which does
not comply with these terms.

Adaptive individual residential load forecasting based on deep learning and dynamic mirror descent

Fujia Han* and Xiaohui Wang

Department of Artificial Intelligence Application, China Electric Power Research Institute, Beijing, China

With a growing penetration of renewable energy generation in the modern power networks, it has become highly challenging for network operators to balance electricity supply and demand. Residential load forecasting nowadays plays an increasingly important role in this aspect and facilitates various interactions between power networks and electricity users. While numerous research works have been proposed targeting at aggregate residential load forecasting, only a few efforts have been made towards individual residential load forecasting. The issue of volatility of individual residential load has never been addressed in forecasting. Thus, to fill this gap, this paper presents a deep learning method empowered with dynamic mirror descent for adaptive individual residential load forecasting. The proposed method is evaluated on a real-life Irish residential load dataset, and the experimental results show that it improves the prediction accuracy by 9.1% and 11.6% in the aspects of RMSE and MAE respectively in comparison with a benchmark method.

KEYWORDS

deep learning, dynamic mirror descent, interactions, renewable energy generation, residential load forecasting

1 Introduction

As advanced metering infrastructure (AMI) is being widely deployed in the modern power system, especially, smart meters, a growing number of granular data of residential electricity consumption has become easily available on a large scale (Sajjad et al., 2016; Xie et al., 2018). This huge amount of data enables power network operators to motivate residential customers to actively participate in demand side management (DSM) through a wide range of various demand response programs (DRPs), for example, time-of-use pricing (Zhou et al., 2016; Ponocko and Milanovic, 2018). As part of DSM, residential load forecasting is a significantly important but challenging task for power network operators, due to great irregularity and uncertainty of residential load (Welikala et al., 2019). As a result, addressing the challenges of residential load forecasting plays a crucial role in interactions between network operators and residential customers, efficient and cost-effective grid operations, and household energy consumption optimizations.

At present, residential load forecasting is generally categorized into two classes - aggregate and individual. More specifically, data analytics for aggregate residential load forecasting mainly include support vector regression (SVR) (Humeau et al., 2013; Wijaya et al., 2015), random forest (Goehry et al., 2020), artificial neural networks (ANNs) (Marinescu et al., 2013; Marinescu et al., 2014; Quilumba et al., 2015; Campos and Silva, 2016; Stephen et al., 2017; Wang et al., 2018; Oprea and Bara, 2019), and deep neural networks (DNNs) (Zheng et al., 2018; Zou et al., 2019). Besides, these methods tend to be combined with clustering techniques, for instance, k-means clustering, in order to improve the forecasting performance. In general, a number of models based on these methods have obtained a desirable level of prediction accuracy on aggregate residential load forecasting. This is because a variety of behaviours of residential customers can smoothen out their overall load profile at the aggregate level, therefore generating an easily identifiable energy consumption pattern.

However, compared to aggregate residential load forecasting, only a few researchers have attempted to explore individual residential load forecasting so far. Some traditional machine learning methods, for instance, ANNs, are still applied to forecast individual residential load (Paterakis et al., 2016; Xu et al., 2016; Vossen et al., 2018; Dinesh et al., 2019; Wu et al., 2020). During recent years, DNNs, such as recurrent neural networks (RNNs) and long short-term memory (LSTM) networks (Gan et al., 2017; Kong et al., 2017; Hossen et al., 2018; Shi et al., 2018; Kong et al., 2019; Alhussein et al., 2020; Wang et al., 2020; Lin et al., 2021), have been largely adopted due to their superior capability in extracting complex patterns. Although existing DNN models have mostly achieved a comparatively higher prediction accuracy than many traditional machine learning models, they are essentially trained on a limited amount of residential load data offline and then applied to perform forecasting online. As a result, these well-trained offline models are likely to encounter many sudden changes of residential load when forecasting online, which are not included in training. This is because individual residential load can be extremely volatile and uncertain, which would have a significantly negative effect on the forecasting performance of the DNN models.

To address the issue of volatility of individual residential load in forecasting, this paper presents a method for short-term individual residential load forecasting, which is able to adjust the forecasting error dynamically. Specifically, the key contributions of the paper are summarized as follows:

- 1) Firstly, it presents an LSTM based deep learning method empowered with dynamic mirror descent (DMD) for adaptive individual residential load forecasting.
- 2) Secondly, it modifies the original DMD to make it feasible for adjusting residential load forecasting.
- 3) Thirdly, it devises a comprehensive feature expression strategy to describe load characteristics at each time step in order to form the input of the forecasting model.
- 4) Finally, the proposed method is validated and compared with a published benchmark method on a real-life Irish residential load dataset, and the influence of the modified DMD on the forecasting performance of the proposed method is investigated in detail.

The remainder of this paper is organized as follows. Section 2 reports a comprehensive literature review on residential load forecasting. Section 3 briefly introduces recurrent neural networks and LSTM networks, and then details the modifications made on the original DMD. Section 4 integrates DMD into an LSTM based deep learning method for adaptive individual residential load forecasting. In Section 5, the proposed residential load forecasting method is evaluated on a real-life Irish residential load dataset. Finally, Section 6 concludes the paper and points out some future work.

2 Review on residential load forecasting

A number of research works have been presented in the area of aggregate residential load forecasting. Wijaya et al. (2015) designed a short-term cluster-based aggregate residential load forecasting strategy. It firstly clusters residential customers, then forecasts the energy consumption of each cluster separately through SVR, and finally aggregates the energy consumption forecasts of all clusters. Similar to Wijaya et al. (2015), Humeau et al. (2013) developed a residential load forecasting method for the district level, which combines k-means clustering with SVR. Different from Wijaya et al. (2015) and Humeau et al. (2013), Goehry et al. (2020) employed hierarchical clustering and random clustering respectively to divide residential customers into subsets and applied random forests to build the forecasting model for each subset.

ANNs have also been commonly applied to forecast residential load at the aggregate level. For example, a dynamic forecasting mechanism was proposed to monitor small-scale residential electricity demand and detect anomalous pattern changes in Marinescu et al. (2014). A self-organizing map is employed for anomalous day detection, and an ANN prediction model changes its input neurons according to a previously detected and recorded match in a database of anomalous days in order to conduct demand prediction for anomalous days. Wang et al. (2018) proposed an ensemble method for short-term aggregate residential load forecasting, which produces the forecasts for all load subprofiles based on hierarchical clustering and ANNs, and then combines all the forecasts with different weights to obtain the final forecasting result of

the aggregate residential load. Quilumba et al. (2015) presented a three-step aggregate residential load forecasting approach, based on k-means clustering and ANNs. In Marinescu et al. (2013) and Campos and Silva (2016), a comprehensive performance comparison was made between ANNs and some other prediction models, such as auto-regression and auto-regressive integrated moving average. Moreover, a gated recurrent unit (GRU) neural network based approach was developed to perform short-term load forecasting for residential community (Zheng et al., 2018). Also, it uses least absolute shrinkage and selection operator (LASSO) and partial correlation analysis to explore the influences of temperature, humidity, rainfall, and wind speed on residential load in order to determine the input variables for the forecasting model.

In summary, aggregate residential load is comparatively easier to forecast than individual residential load, because individual behaviors have the features of volatility and uncertainty in energy consumption. As a result, existing forecasting models on aggregate residential load have obtained a satisfactory level of prediction accuracy.

By contrast, only a few efforts have been made towards individual residential load forecasting. Xu et al. (2016) proposed a k-nearest vector auto-regressive framework with exogenous input to spatial-temporally model household electricity demand. Dinesh et al. (2019) presented a forecasting approach to the power consumption of a single household, which is based on non-intrusive load monitoring (NILM) and graph spectral clustering. Different from Xu et al. (2016) and Dinesh et al. (2019), Vossen et al. (2018) developed a probabilistic forecasting model to describe the uncertainty of individual residential load using two different types of density-estimation ANNs respectively. In Wu et al. (2020), a boosting-based framework for multiple kernel learning regression was presented to forecast individual residential load. It not only adopts boosting to learn an ensemble of multiple kernel regressors, but also applies transfer learning to forecast the load of the residential customer which has a very limited amount of energy consumption data. In addition, Gan et al. (2017) employed a quantile LSTM network to perform probabilistic residential load forecasting at the individual level. In Alhussein et al. (2020), a hybrid model combining a convolutional neural network and an LSTM network was proposed to forecast the individual household load. Wang et al. (2020) designed a framework for short-term individual residential load forecasting. It firstly partitions historical load data by clustering to train multiple LSTM models, and then uses a fully connected cascade neural network to fuse the multiple LSTM models. Shi et al. (2018) proposed a novel pooling-based deep RNN to avoid overfitting during residential load forecasting. It batches the load profiles of a group of

residential customers into an input pool in order to increase data diversity and volume. Also, Lin et al. (2021) presented a graph neural network based method for individual residential load forecasting, which aims to capture both temporal information of historical load and spatial information of neighbouring households in order to improve the forecasting accuracy. In Hossen et al. (2018), different types of DNNs, such as RNNs and LSTM networks, were applied to short-term individual residential load forecasting and a performance comparison was conducted among them. Furthermore, automatic hyperparameter tuning was utilised to select an optimal hyperparameter combination for an LSTM network in order to improve the accuracy of individual residential load forecasting (Kong et al., 2017).

Although a variety of forecasting models for individual residential load have been developed, their training data is unable to include all the cases on residential energy consumption, as individual residential load tends to change dramatically over time, which leads to a poor prediction accuracy when they are applied online.

3 Methodology

3.1 Long short term memory model

As a sequence based model, RNNs are capable of establishing excellent temporal correlation between previous and current information (Chen et al., 2016; Tolosana et al., 2018). This characteristic makes RNNs an ideal candidate for short-term residential load forecasting, because the residential load consumption pattern has a strong and complex relationship between adjacent time steps (Kong et al., 2019). However, in terms of the specific implementation, a special RNN, called the LSTM network, is employed in this paper, as it significantly improves the performance of the general RNN. In this section, the RNN architecture is firstly introduced, and then the LSTM unit is explained.

3.1.1 Recurrent neural networks

In the working process, the RNN aims to map the input sequence of x values into corresponding sequential outputs: y . Specifically, the learning process conducts every single time step from $t = 1$ to $t = \tau$. For time step t , the network neuron parameters at the l^{th} layer update their shared states with the following equations (Shi et al., 2018):

$$a_1^{(t)} = b_1 + W_1 \cdot h_1^{(t-1)} + U_1 \cdot x^{(t)} \quad (1)$$

$$h_l^{(t)} = f_{\text{activation}}(a_l^{(t)}) \quad (l = 1, 2, \dots, N) \quad (2)$$

$$a_l^{(t)} = b_l + W_l \cdot h_l^{(t-1)} + U_l \cdot h_{l-1}^{(t)} \quad (l = 2, 3, \dots, N) \quad (3)$$

$$y^{(t)} = b_N + W_N \cdot h_N^{(t-1)} + U_N \cdot h_N^{(t)} \quad (4)$$

$$L = f_{\text{loss}}(y^{(t)}, y_{\text{target}}^{(t)}) \quad (5)$$

where $x^{(t)}$ is the data input at time step t , $y^{(t)}$ is the corresponding forecast, $y_{\text{target}}^{(t)}$ is the true output at time step t , $h_l^{(t)}$ is the shared states of the l^{th} layer at time step t , N is the total layer number of the network, and $a_l^{(t)}$ is the input of the l^{th} layer at time step t , which consists of three components: 1) the input $x^{(t)}$ at time step t or the shared states $h_{l-1}^{(t)}$ of the $l - 1^{\text{th}}$ layer at time step t ; 2) the bias b_l of the l^{th} layer; 3) the shared states $h_l^{(t-1)}$ of the l^{th} layer at time step $t-1$.

Due to their shared states, RNNs are able to learn dependency contained in the previous time steps.

3.1.2 LSTM units

RNNs are trained by backpropagation through time, but learning long-term dependency with RNNs is difficult because of gradient vanishing or exploding (Kong et al., 2019). In order to overcome these two issues, an LSTM unit is introduced, and LSTM has gradually become the most popular structure of RNNs in solving many time series problems.

Let $\{x_1, x_2, \dots, x_T\}$ denote a typical input sequence for an LSTM unit, where $x_t \in R^k$ represents a k -dimensional vector of real values at time step t . In order to establish temporal relations, the LSTM unit defines and maintains an internal memory cell state throughout the life cycle, which is the most important element of the LSTM unit. The memory cell state s_{t-1} interacts with the intermediate output h_{t-1} and the subsequent input x_t to determine which elements of the internal state vector should be updated, maintained, or erased according to the outputs of the previous time step and the inputs of the present time step. Apart from the internal state, the LSTM unit also defines the input node g_t , the input gate i_t , the forget gate f_t , and the output gate o_t . The formulations of all nodes in the LSTM unit are presented below from (6) to (11) (Kong et al., 2019):

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \quad (6)$$

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \quad (7)$$

$$g_t = \Phi(W_{gx}x_t + W_{gh}h_{t-1} + b_g) \quad (8)$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \quad (9)$$

$$s_t = g_t \odot i_t + s_{t-1} \odot f_t \quad (10)$$

$$h_t = \Phi(s_t) \odot o_t \quad (11)$$

where W_{fx} , W_{fh} , W_{ix} , W_{ih} , W_{gx} , W_{gh} , W_{ox} , and W_{oh} denote the weight matrices of the corresponding inputs of the network activation functions, \odot denotes the element-wise multiplication, σ denotes the sigmoid activation function, and Φ denotes the tanh activation function.

In each time step, the memory cell state has three operations: 1) discard useless information from the memory cell state s_t ; 2) add the new information i_t extracted from the input x_t and the intermediate output h_{t-1} into the memory cell state s_t ; 3) determine the new

intermediate output h_t from the memory cell state s_t . Thus, the memory cell state is capable to keep useful information for a long time and result in RNN performance enhancement.

3.2 Dynamic mirror descent

As an online learning method, dynamic mirror descent (DMD) is capable to incorporate a dynamic model in the learning process, and effectively minimize the loss and estimate time-varying system states (Hall and Willett, 2015; Ledva et al., 2015). DMD is executed by two main steps: 1) an observation-based update incorporates the new measurement into the parameter prediction; 2) a model-based update advances the parameter prediction to the next time step. The frequently used notations in DMD are given in Table 1, and the detailed steps of DMD are presented in Algorithm 1.

Input: Data point x
Output: Final forecast $\hat{\theta}$
1 Given a non-increasing sequence of step size $\eta_t > 0$, initialize $T > 0, t = 1, \hat{\theta}_1 \in \Theta$
2 For $t \leq T$ Do
3 Observe x_t and incur loss $l_t(\hat{\theta}_t)$
4 Environment produces a dynamic model Φ_t
5 Calculate $\check{\theta}_{t+1} = \arg \min_{\theta \in \Theta} \langle \nabla f_t(\hat{\theta}_t), \theta \rangle + \eta_t r(\theta) + D(\theta \hat{\theta}_t)$
6 Calculate $\hat{\theta}_{t+1} = \Phi_t(\check{\theta}_{t+1})$
7 $t = t + 1$
8 End
9 Terminate

Algorithm 1: Dynamic Mirror Descent

In order to apply DMD to adjust the forecasted value of residential load dynamically, following the work presented in Ledva et al. (2018), a few modifications are made to the original DMD. The idea is that the concept of the original DMD is still adopted but it is not a direct implementation of the original DMD. In other words, the modified DMD considers the forecasting model as a black box and simply adjusts its output with the measured and forecasted values. Hence, the modified DMD is formulated as follows:

$$\hat{k}_{t+1} = \arg \min_{\theta \in \Theta} \eta \langle \nabla l_t(\hat{\theta}_t), \theta \rangle + D(\theta || \hat{k}_t) \quad (12)$$

$$\check{\theta}_{t+1} = \Phi(\hat{\theta}_t) \quad (13)$$

$$\hat{\theta}_{t+1} = \check{\theta}_{t+1} + \hat{k}_{t+1} \quad (14)$$

where ∇l_t is an arbitrary sub-gradient function of $l_t(\cdot)$; \hat{k}_t is the adjustment variable accumulating the deviation between the forecasted and measured values; η is the constant step size; Φ is the residential load forecasting model; $\hat{\theta}_t$ is the input data of Φ . The model-based update (13) only computes an intermediate prediction $\check{\theta}_{t+1}$ without the real measurement influencing $\check{\theta}_{t+1}$. The measurement-based update and the model-based prediction are combined in (14) to obtain a final prediction $\hat{\theta}_{t+1}$.

TABLE 1 Frequently used notations in dynamic mirror descent.

Notation	Meaning
θ	the system state
Θ	a convex set
x_t	the observed data point
η_t	the step size
Φ_t	a known dynamic model based on prior knowledge
θ_t	the intermediate forecast
$\hat{\theta}_t$	the final forecast
$f_t(\cdot)$	a convex function from the environment
$l_t(\cdot)$	a convex loss function
$r(\cdot)$	a convex regularization function
$\nabla f_t(\cdot)$	an arbitrary subgradient function of $f_t(\cdot)$
$\langle \cdot, \cdot \rangle$	the dot product operator
$D(\theta \ \hat{\theta}_t)$	the Bregman divergence measuring the distance between θ and $\hat{\theta}_t$

In this paper, for simplicity, the loss function $l_t(\hat{\theta}_t)$ is selected as $l_t(\hat{\theta}_t) = \frac{1}{2} \|\hat{\theta}_t - y_t\|_2^2$, while the Bregman divergence $D(\theta \|\hat{k}_t)$ is selected as $D(\theta \|\hat{k}_t) = \frac{1}{2} \|\theta - \hat{k}_t\|_2^2$. Thus, the convex function (12) can be simplified as the following:

$$\hat{k}_{t+1} = \hat{k}_t + \eta(y_t - \hat{\theta}_t) \tag{15}$$

where y_t and $\hat{\theta}_t$ are the real measurement and the final forecast of residential load respectively. As a result, the modified DMD is formed using (13–15).

4 Adaptive individual residential load forecasting

4.1 Implementation process

Due to high volatility and uncertainty of individual residential load, a comprehensive feature expression strategy is devised in order to describe the details of the energy consumption at each time step. So, the input features of a data sample S_t at a particular time step t are detailed as follows:

- 1) the sequence of the residential load for the past T time steps $E_t \in \mathbf{R}^T$ is formed as:

$$E_t = [e_{t-T}, \dots, e_{t-2}, e_{t-1}] \tag{16}$$

where e_t is the energy consumption (kWh) at time step t ;

- 2) the sequence of the half-hourly indexes for the past T time steps $D_t \in \mathbf{R}^T$ is formed as:

$$D_t = [d_{t-T}, \dots, d_{t-2}, d_{t-1}] \tag{17}$$

where $d_t \in [1, 48]$ is the half-hourly index for time step t , because the sampling frequency is once every half an hour;

- 3) the sequence of the day indexes for the past T time steps $W_t \in \mathbf{R}^T$ is formed as:

$$W_t = [w_{t-T}, \dots, w_{t-2}, w_{t-1}] \tag{18}$$

where $w_t \in [1, 7]$ is the day index for time step t , as there are 7 days in a week;

- 4) the sequence of the holiday signs for the past T time steps $H_t \in \mathbf{R}^T$ is formed as:

$$H_t = [h_{t-T}, \dots, h_{t-2}, h_{t-1}] \tag{19}$$

where h_t is the holiday sign for time step t , which is either one or 2, and one denotes non-holiday and two denotes holiday (in this paper, it is assumed that weekdays are non-holiday and weekends are holiday).

Thus, a data sample S_t is a matrix of a concatenation of the four sequences, expressed below:

$$S_t = [E_t^T, D_t^T, W_t^T, H_t^T] \tag{20}$$

where E_t^T, D_t^T, W_t^T , and H_t^T are the transposes of E_t, D_t, W_t , and H_t respectively. In order to speed up the convergence of the forecasting model and improve its generalization capacity, the input features are normalized to [0, one] according to their nature. To be specific, the min-max normalization method is adopted for E_t , while D_t, W_t , and H_t are encoded by a one-hot encoder. The one-hot encoder maps an original element of the feature sequence with M categories into a new sequence with M elements, where only the new element corresponding to the original element is one while the rest are all zeros. Hence, a normalized data sample \hat{S}_t is expressed as:

$$\hat{S}_t = \left[\widehat{E}_t^T, \widehat{D}_t^T, \widehat{W}_t^T, \widehat{H}_t^T \right] \tag{21}$$

where \hat{S}_t is a $T \times N$ matrix and $N = 1 + 48 + 7 + 2 = 58$, $\widehat{E}_t^T, \widehat{D}_t^T, \widehat{W}_t^T$, and \widehat{H}_t^T are the normalized matrixes of E_t^T, D_t^T, W_t^T , and H_t^T respectively. Each row of the normalized data sample \hat{S}_t is the detailed features for the corresponding time step.

In order to perform adaptive residential load forecasting, an LSTM network is firstly well trained for each resident, and then it is applied with the modified DMD adjusting the forecasting error dynamically. In general, the proposed method goes through the following four steps sequentially to forecast residential load: 1) the input sample is formed; 2) the input sample is normalized and fed to the well trained LSTM network to obtain the intermediate forecast; 3) the adjustment variable of the modified DMD is updated; 4) the final forecast is computed by summing the adjustment variable and the intermediate forecast. The framework of integration of deep learning and dynamic mirror descent for adaptive individual residential load forecasting is shown in Figure 1, and the steps of LSTM and DMD integration are detailed in Algorithm 2.

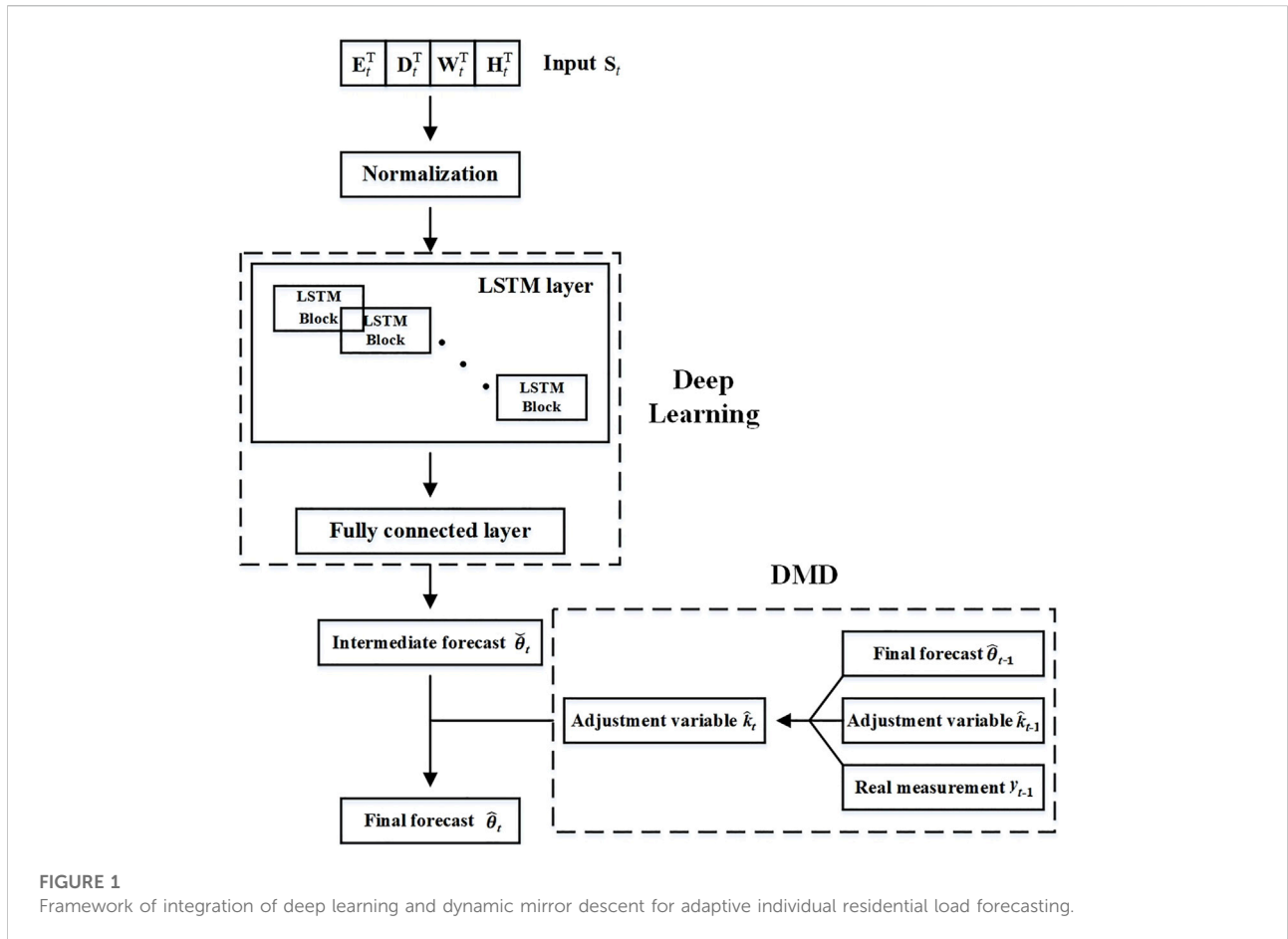


FIGURE 1 Framework of integration of deep learning and dynamic mirror descent for adaptive individual residential load forecasting.

Input: Residential load sample S
Output: Final residential load forecast $\hat{\theta}$
1 Select and preprocess residential load data
2 Generate a training dataset Ψ_{tr} and a test dataset Ψ_{ts}
3 Normalize all samples in the training dataset Ψ_{tr} using (16) – (21)
4 Set values for the hyperparameters for an LSTM network
5 Train the LSTM network on the training dataset Ψ_{tr}
6 Obtain a well-trained forecasting model Φ
7 Initialize the step size η , the test dataset size T_{max} , and the adjustment variable \hat{k}_1 , $t = 1$
8 For $t \leq T_{max}$ Do
9 Normalize the sample S_t of the test dataset Ψ_{ts} using (16) – (21)
10 Input the normalized sample \hat{S}_t into the forecasting model Φ to obtain $\hat{\theta}_t$ using (13)
11 Update \hat{k}_t using (15)
12 Calculate $\hat{\theta}_t$ using (14)
13 $t = t + 1$
14 End
15 Evaluate the prediction performance on the test dataset Ψ_{ts}
16 Terminate

TABLE 2 Experiment parameters of adaptive individual residential load forecasting based on deep learning and dynamic mirror descent.

Parameters	Values
RNN layer number	3
Fully-connected layer number	1
Neuron number of RNN layer	64
Neuron number of fully-connected layer	64
Batch size	128
Time length of input	24
RNN unit	LSTM
Activation function of RNN layer	tanh
Activation function of fully-connected layer	linear
Optimization method	AdamOptimizer
Training epoch	25
Learning rate	0.001
Loss function	RMSE

4.2 Dataset description

The dataset used in this paper is from the Smart Metering Electricity Customer Behaviour Trials initiated by Commission

for Energy Regulation in Ireland (Commission for Energy Regulation, 2012). The trials lasted from July 2009 to December 2010 with over 5000 Irish residential customers and small and medium enterprises (SMEs) participating. In the trials, there are 929 1-E-E customers, which means that they are all residential customers (1) with the controlled stimulus (E) and the controlled tariff (E). These customers are billed at the flat rate without any stimulus, and therefore are most representative because the majority of residential customers outside the trials are of this type. Among these 929 customers, 782 customers have a complete record of energy consumption throughout the trials. In this paper, 750 1-E-E customers with a complete record are randomly selected as the experiment dataset.

4.3 Experiment setup

The full data of a single residential customer is divided into a training dataset and a test dataset with a ratio of 9:1. So, for each resident, 90% of the data samples are used for training, while the rest of 10% are used for testing. In addition, as this paper is not focused on improving the prediction accuracy via the optimal network structure, hyperparameter fine-tuning is not conducted on the LSTM network. All the experiment parameters are presented in Table 2.

TABLE 3 Residential load forecasting results of the proposed and benchmark methods.

Index (kWh)	Benchmark Method	Proposed Method	Improvement percentage (%)
MAE	0.285	0.252	11.6%
RMSE	0.493	0.448	9.1%

5 Results and discussion

In this section, a performance comparison was firstly made between the proposed residential load forecasting method and a published benchmark method presented in Kong et al. (2019). It is noted that the benchmark method only uses the same LSTM network as the proposed method but does not apply any online learning method. RMSE and MAE are employed as the performance indexes for residential load forecasting, formulated as follows:

$$RMSE = \sqrt{\frac{\sum_{t=1}^N (\hat{y}_t - y_t)^2}{N}} \tag{22}$$

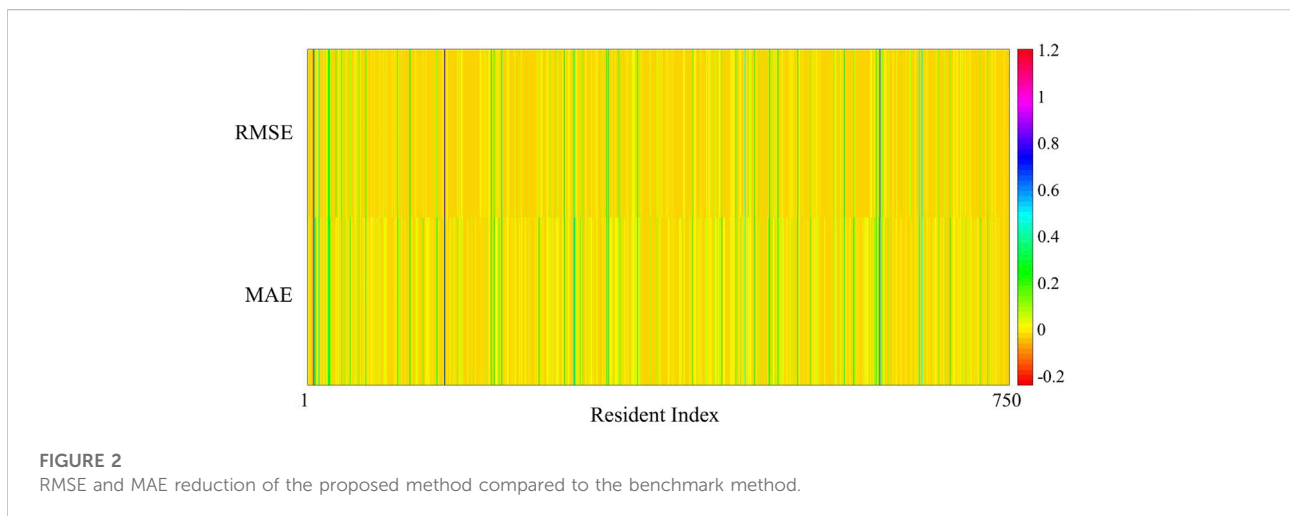
$$MAE = \frac{\sum_{t=1}^N |\hat{y}_t - y_t|}{N} \tag{23}$$

where \hat{y}_t is the forecasted value, y_t is the real value, and N is the size of the test dataset. Furthermore, the effect of the parameter η of the modified DMD on the proposed residential load forecasting method was investigated. In this paper, the adjustment variable is initialised as 0.

5.1 Performance analysis of adaptive individual residential load forecasting

A performance comparison was conducted between the proposed and benchmark methods in terms of prediction accuracy. In this case, the parameter η of the modified DMD is set as 1.0×10^{-5} , 1.0×10^{-4} , 1.0×10^{-3} , 1.0×10^{-2} , 1.0×10^{-1} , and 1.0×10^0 respectively, and the optimal forecasting result obtained is regarded as the result of the proposed method. The results of both methods are presented in Table 3.

It is noted that Table 3 describes the average RMSE and MAE of all the residents. In Table 3, the proposed method performs



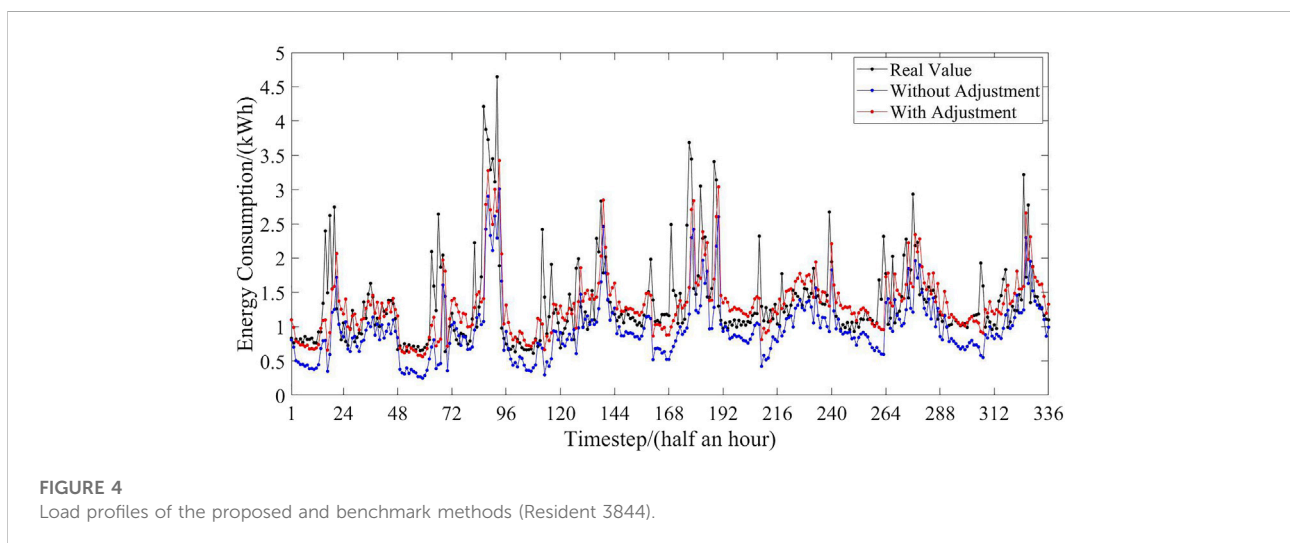
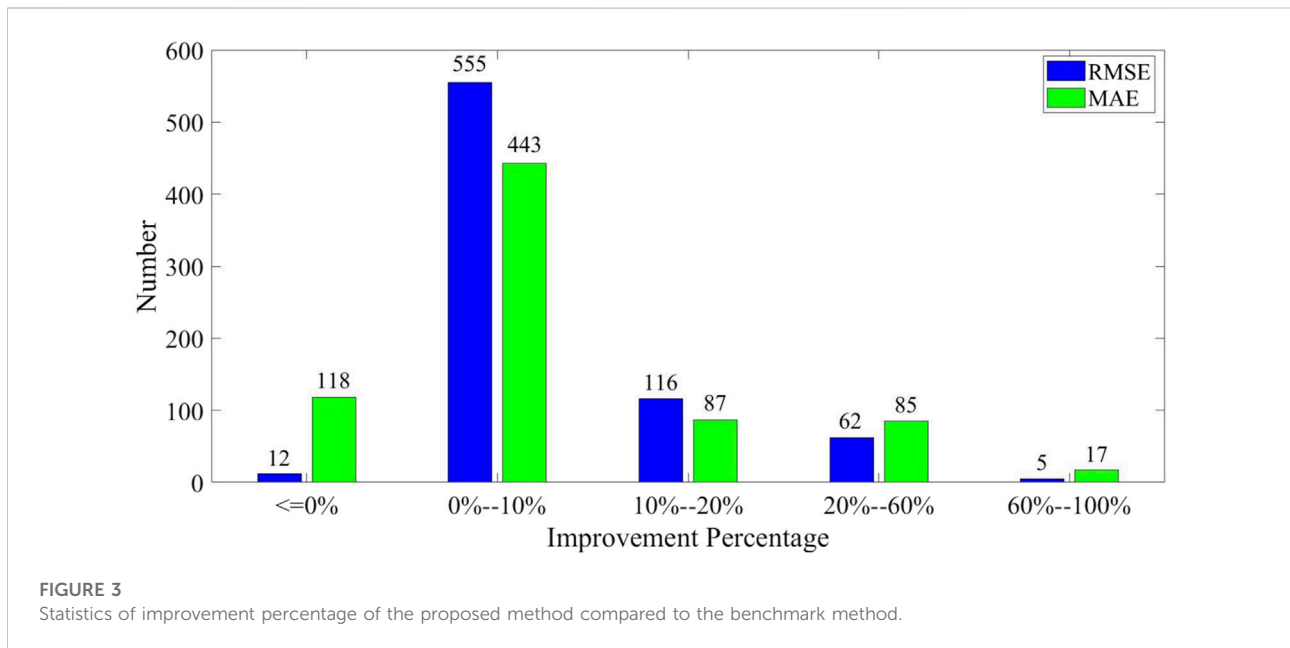
much better than the benchmark method, in terms of both RMSE and MAE. Besides, the improvement percentage of MAE is higher than that of RMSE, because MAE and RMSE indicate the forecasting performance from two different perspectives. To be specific, MAE, which reflects the mean of errors, regards every error equally and averages all the errors, while RMSE, which reflects the fluctuation of errors, strengthens the large error and weakens the small error.

The reason for the significant performance improvement of the proposed method can be explained as follows. As the adjustment variable \hat{k}_t of the modified DMD is capable to update itself based on the errors between the forecasts and the

real measurements of the previous time steps, the proposed method can effectively adjust the intermediate forecast $\hat{\theta}_t$ of the current time step to obtain the final forecast $\hat{\theta}_t$.

In addition, Figure 2 presents the RMSE and MAE reduction of the proposed method across all the residents compared to the benchmark method, while Figure 3 presents the statistics of improvement percentage of the proposed method compared to the benchmark method.

In Figure 2, it can be clearly seen that the proposed method achieves different levels of improvements on a large number of residents. More specifically, some residents receive significant RMSE and MAE reductions



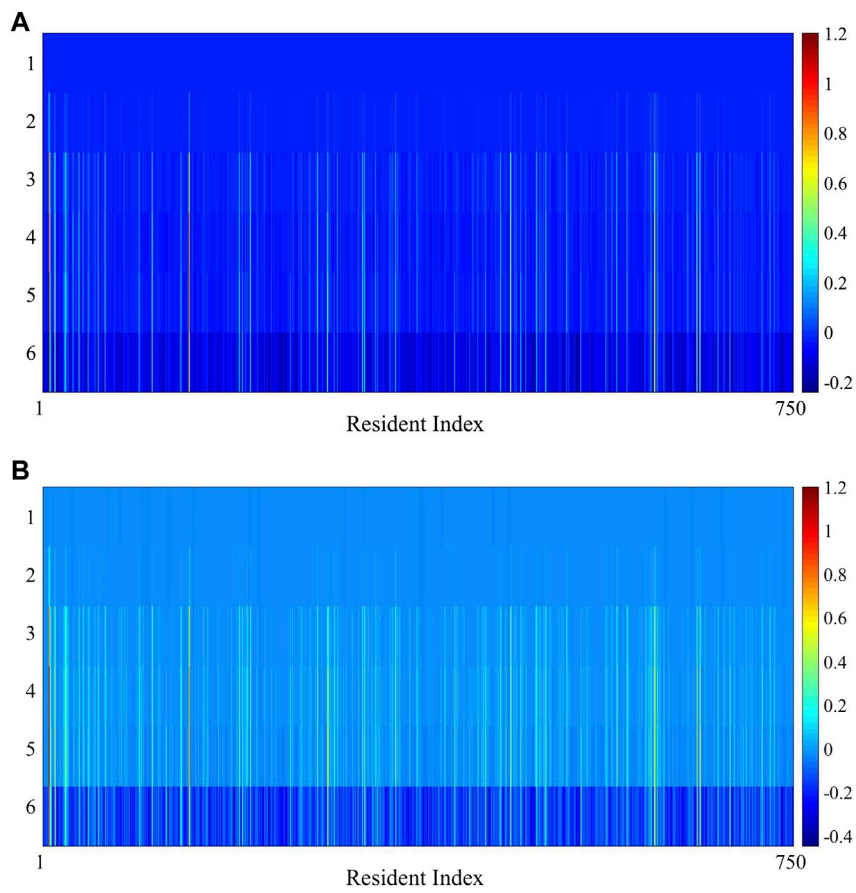


FIGURE 5 Effect of parameter η on the forecasting performance of the proposed method: 1) $\eta = 1.0 \times 10^{-5}$, 2) $\eta = 1.0 \times 10^{-4}$, 3) $\eta = 1.0 \times 10^{-3}$, 4) $\eta = 1.0 \times 10^{-2}$, 5) $\eta = 1.0 \times 10^{-1}$, 6) $\eta = 1.0 \times 10^0$. **(A)** MAE reduction. **(B)** RMSE reduction.

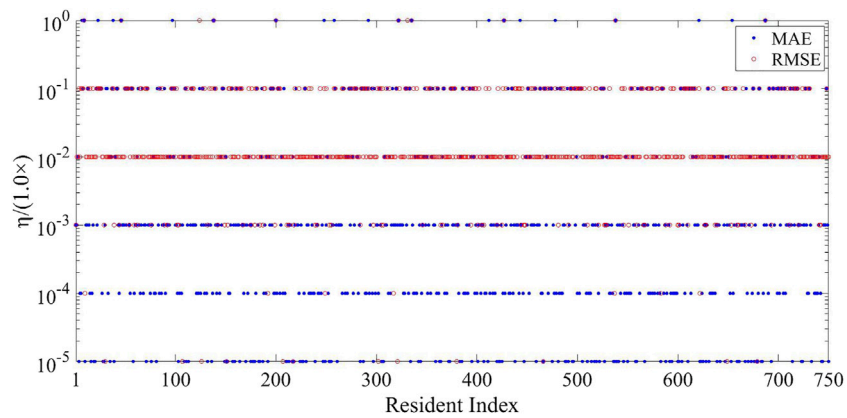


FIGURE 6 Optimal values of parameter η for the proposed method to achieve the minimum RMSE and MAE values.

(e.g., from 0.4 to 1.0), but others only receive slight RMSE and MAE reductions (e.g., from 0.01 to 0.05). It is also noted that some residents obtain an RMSE decrease but an MAE increase, while others obtain the opposite result. Besides, there is no performance difference between the proposed and benchmark methods for a few residents. This demonstrates that the modified DMD fails to effectively adjust the forecasting error over time, mainly because of the great complexity of these residential load profiles.

In Figure 3, there are totally $555 + 116 + 62 + 5 = 738$ residents with an RMSE reduction, which account for $738/750 = 98.4\%$ of all residents. Among them, most residents obtain an RMSE reduction of less than 20%, which account for

$(555 + 116)/750 = 89.47\%$. Besides, only five residents obtain an RMSE reduction of even more than 60%. Similarly, a total of $443 + 87 + 85 + 17 = 632$ residents receive an MAE reduction, which account for $632/750 = 84.27\%$. Among them, most residents obtain an MAE reduction of less than 20%, which account for $(443 + 87)/750 = 70.67\%$. Only 17 residents obtain an MAE reduction of even more than 60%. It is also noted that 118 residents fail to obtain an MAE reduction, while only 12 residents fail to obtain an RMSE reduction. This fact indicates that the proposed method tends to decrease RMSE in comparison with MAE.

Furthermore, Figure 4 shows the load profiles of a random resident forecasted by the proposed and benchmark methods

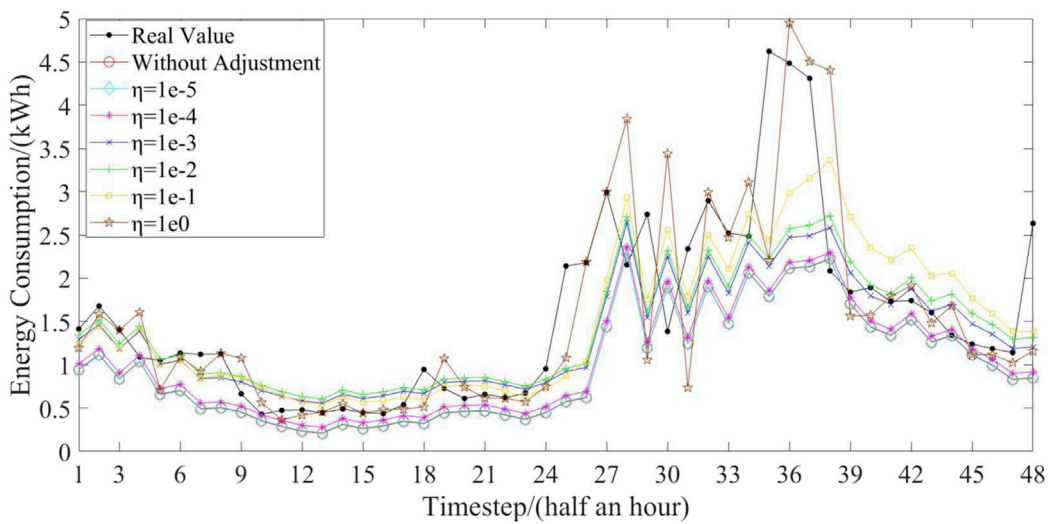


FIGURE 7 Load profiles of the proposed method for different values of parameter η (Resident 1425).

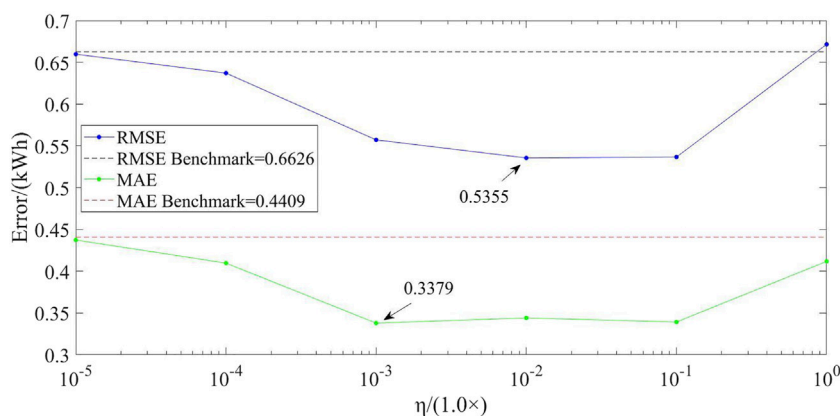


FIGURE 8 Forecasting errors of the proposed method for different values of parameter η (Resident 1425).

during a random week (Monday 29/11/2010—Sunday 5/12/2010). It is obvious in Figure 4 that the forecasted load profile of the proposed method is much closer to the real load profile than that of the benchmark method. To be specific, when a dramatic increase or decrease of the residential load occurs, the proposed method can capture the change rapidly. Also, it can track the residential load stably, when the residential load only fluctuates slightly. By contrast, the benchmark method is unable to forecast accurately, when the residential load changes significantly over time.

5.2 Effect of parameter η on performance of adaptive individual residential load forecasting

As the parameter η plays an important role in the modified DMD, its effect on the forecasting performance of the proposed method was further investigated. Figure 5 depicts the MAE and RMSE reduction of the proposed method across all residents compared to the benchmark method, when the modified DMD is applied with different values of the parameter η .

It can be clearly seen in Figure 5A that the proposed method has the worst performance when η is 1.0×10^0 , because a large number of residents fail to receive an MAE reduction. However, when η is 1.0×10^{-1} , 1.0×10^{-2} , and 1.0×10^{-3} , the proposed method performs much better, because a majority of residents receive different MAE reductions. Only a small amount of residents receive an MAE reduction, when η is 1.0×10^{-4} and 1.0×10^{-5} . It is also noted that there are a variety of trends of MAE reductions among all the residents as η changes from 1.0×10^0 to 1.0×10^{-5} . Likewise, in Figure 5B, in terms of RMSE, the proposed method has the worst performance when η is 1.0×10^0 , but performs much better when η is 1.0×10^{-1} , 1.0×10^{-2} , and 1.0×10^{-3} . Only a small number of residents obtain an RMSE reduction, when η is 1.0×10^{-4} and 1.0×10^{-5} . Besides, different trends of RMSE reductions can be seen among all the residents, as η changes from 1.0×10^0 to 1.0×10^{-5} .

The reason for the poor forecasting performance of the proposed method, when η is too small or too large, can be explained as follows. In (15), if η is too small, the deviation between the real measurement and the final forecast at the current time step cannot be accumulated effectively in the adjustment variable at the next time step. Thus, the modified DMD is unable to adjust the intermediate forecast properly over time. But, if η is too large, the deviation between the real measurement and the final forecast at the current time step accounts for a large proportion in the adjustment variable at the next time step, significantly weakening the accumulation of the deviations of the previous time steps. Therefore, the modified DMD fails to track the forecasting error accurately over time.

Furthermore, Figure 6 shows the optimal values of the parameter η when the proposed method achieves the best performance in terms of RMSE and MAE respectively. It is obvious in Figure 6 that the proposed method rarely achieves the minimum MAE value when η is 1.0×10^0 and 1.0×10^{-5} . However, it is capable to achieve the minimum MAE value on a large amount of residents when η is 1.0×10^{-3} and 1.0×10^{-4} . Similarly, the proposed method is able to achieve the minimum RMSE value only on a few residents when η is 1.0×10^0 and 1.0×10^{-5} . But, it achieves the minimum RMSE value on a majority of residents when η is 1.0×10^{-1} and 1.0×10^{-2} . As MAE and RMSE measure errors from two perspectives, the optimal values of η are different, when the proposed method achieves the minimum MAE and RMSE values on a single resident.

In addition, Figure 7 describes the load profiles of a random resident forecasted by the proposed method on a random day (Wednesday 22/12/2010) when the modified DMD is applied with different values of the parameter η , while Figure 8 describes the forecasting errors of the proposed method on this resident as the parameter η changes.

In Figure 7, the proposed method performs forecasting accurately at most time steps throughout the day, when η is 1.0×10^{-2} and 1.0×10^{-3} . By contrast, when η is 1.0×10^{-4} and 1.0×10^{-5} , the forecasted load profiles of the proposed method are quite close to that of the benchmark method. When η is 1.0×10^0 and 1.0×10^{-1} , there is a significant deviation between the real load profile and the forecasted load profiles of the proposed method at many time steps. This is mainly because a too small or too large value of η has a negative influence on the modified DMD. In Figure 8, MAE of the proposed method firstly decreases from $\eta = 1.0 \times 10^{-5}$ to $\eta = 1.0 \times 10^{-3}$, and then increases from $\eta = 1.0 \times 10^{-3}$ to $\eta = 1.0 \times 10^0$. The proposed method achieves the lowest MAE value of 0.3379 when η is 1.0×10^{-3} . Similarly, RMSE of the proposed method firstly decreases from $\eta = 1.0 \times 10^{-5}$ to $\eta = 1.0 \times 10^{-2}$, and then increases from $\eta = 1.0 \times 10^{-2}$ to $\eta = 1.0 \times 10^0$. The proposed method achieves the lowest RMSE value of 0.5355 when η is 1.0×10^{-2} .

6 Conclusion

This paper has presented an adaptive individual residential load forecasting method, which integrates deep learning and dynamic mirror descent to address the issue of great volatility of individual residential load. The original DMD is modified to become feasible for dynamic residential load forecasting. Besides, a detailed feature expression strategy is devised to provide the proposed method with sufficient information of energy consumption at each time step. The experimental results have shown that the proposed method has improved the prediction accuracy substantially by 9.1% in RMSE and 11.6% in MAE, in comparison with the published benchmark method. In addition, the effect of the parameter η of the modified DMD on the proposed method is

further explored, and the comparison results have indicated that the optimal value of η can be found out to achieve the maximum performance improvement.

Future work will focus on fine-tuning techniques to combine with deep learning and explore their effects on residential load forecasting. Optimization techniques will also be applied to search for the optimal value of the parameter η of the modified DMD in a continuous space in our future work.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: <https://www.ucd.ie/issda/data/commissionforenergyregulationcer>.

Author contributions

FH: responsible for investigation, conceptualization, methodology, formal analysis, visualization, original draft preparation, and review and editing. XW: responsible for review and editing, revision, project administration, and funding acquisition.

References

- Alhussein, M., Aurangzeb, K., and Haider, S. I. (2020). Hybrid CNN-LSTM model for short-term individual household load forecasting. *IEEE Access* 8, 180544–180557. doi:10.1109/ACCESS.2020.3028281
- Campos, B. P., and da Silva, M. R. D. (2016). "Demand forecasting in residential distribution feeders in the context of smart grids," in Proceedings of the 2016 12th IEEE International Conference on Industry Applications (INDUSCON), 20–23 November 2016. (Curitiba, Brazil: IEEE). doi:10.1109/INDUSCON.2016.7874464
- Chen, X., Liu, X., Wang, Y., Gales, M. J. F., and Woodland, P. C. (2016). Efficient training and evaluation of recurrent neural network language models for automatic speech recognition. *IEEE/ACM Trans. Audio Speech Lang. Process.* 24 (11), 2146–2157. doi:10.1109/TASLP.2016.2598304
- Commission for Energy Regulation (CER) (2012). Commission for energy regulation (CER). Available at: <https://www.ucd.ie/issda/data/commissionforenergyregulationcer/>.
- Dinesh, C., Makomin, S., and Bajic, I. V. (2019). Residential power forecasting using load identification and graph spectral clustering. *IEEE Trans. Circuits Syst. II* 66 (11), 1900–1904. doi:10.1109/TCSII.2019.2891704
- Gan, D., Wang, Y., Zhang, N., and Zhu, W. (2017). Enhancing short-term probabilistic residential load forecasting with quantile long-short-term memory. *J. Eng.* 2017 (14), 2622–2627. doi:10.1049/joe.2017.0833
- Goehry, B., Goude, Y., Massart, P., and Poggi, J. (2020). Aggregation of multi-scale experts for bottom-up load forecasting. *IEEE Trans. Smart Grid* 11 (3), 1895–1904. doi:10.1109/TSG.2019.2945088
- Hall, E. C., and Willett, R. M. (2015). Online convex optimization in dynamic environments. *IEEE J. Sel. Top. Signal Process.* 9 (4), 647–662. doi:10.1109/JSTSP.2015.2404790
- Hossen, T., Nair, A. S., Chinnathambi, R. A., and Ranganathan, P. (2018). "Residential load forecasting using deep neural networks (DNN)," in Proceedings of the 2018 North American Power Symposium (NAPS), 9–11 September 2018 (Fargo, ND, USA: IEEE). doi:10.1109/NAPS.2018.8600549
- Humeau, S., Wijaya, T. K., Vasirani, M., and Aberer, K. (2013). "Electricity load forecasting for residential customers: Exploiting aggregation and correlation between households," in Proceedings of the 2013 Sustainable Internet and ICT for Sustainability (SustainIT), 30–31 October 2013 (Palermo, Italy: IEEE). doi:10.1109/SustainIT.2013.6685208
- Kong, W., Dong, Z., Jia, Y., Hill, D. J., Xu, Y., and Zhang, Y. (2019). Short-term residential load forecasting based on LSTM recurrent neural network. *IEEE Trans. Smart Grid* 10 (1), 841–851. doi:10.1109/TSG.2017.2753802
- Kong, W., Dong, Z., Luo, F., Meng, K., Zhang, W., Wang, F., et al. (2017). "Effect of automatic hyperparameter tuning for residential load forecasting via deep learning," in Proceedings of the 2017 Australasian Universities Power Engineering Conference (AUPEC), 19–22 November 2017 (Melbourne, VIC, Australia: IEEE). doi:10.1109/AUPEC.2017.8282478
- Ledva, G. S., Balzano, L., and Mathieu, J. L. (2015). "Inferring the behavior of distributed energy resources with online learning," in Proceedings of the 2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton), 29 September - 2 October 2015 (Monticello, IL, USA: IEEE). doi:10.1109/ALLERTON.2015.7447003
- Ledva, G. S., Balzano, L., and Mathieu, J. L. (2018). Real-time energy disaggregation of a distribution feeder's demand using online learning. *IEEE Trans. Power Syst.* 33 (5), 4730–4740. doi:10.1109/TPWRS.2018.2800535
- Lin, W., Wu, D., and Boulet, B. (2021). Spatial-temporal residential short-term load forecasting via graph neural networks. *IEEE Trans. Smart Grid* 12 (6), 5373–5384. doi:10.1109/TSG.2021.3093515
- Marinescu, A., Dusparic, I., Harris, C., Cahill, V., and Clarke, S. (2014). "A dynamic forecasting method for small scale residential electrical demand," in Proceedings of the 2014 International Joint Conference on Neural Networks (IJCNN), 6–11 July 2014 (Beijing, China: IEEE). doi:10.1109/IJCNN.2014.6889425
- Marinescu, A., Harris, C., Dusparic, I., Clarke, S., and Cahill, V. (2013). "Residential electrical demand forecasting in very small scale: An evaluation of forecasting methods," in Proceedings of the 2013 2nd International Workshop on Software Engineering Challenges for the Smart Grid (SE4SG), 18 May 2013 (San Francisco, CA, USA: IEEE). doi:10.1109/SE4SG.2013.6596108
- Oprea, S. V., and Bara, A. (2019). Machine learning algorithms for short-term load forecast in residential buildings using smart meters, sensors and big data solutions. *IEEE Access* 7, 177874–177889. doi:10.1109/ACCESS.2019.2958383
- Paterakis, N. G., Tascikaraoglu, A., Erdinc, O., Bakirtzis, A. G., and Catalao, J. P. S. (2016). Assessment of demand-response-driven load pattern elasticity using a combined approach for smart households. *IEEE Trans. Ind. Inf.* 12 (4), 1529–1539. doi:10.1109/TII.2016.2585122

Funding

This work was funded by State Grid Corporation of China, under Grant Number 5700-20205267A-0-0-00 (Data Mining Technology of Potential High-Value Industrial Users for Data Operations).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

- Ponocko, J., and Milanovic, J. V. (2018). Forecasting demand flexibility of aggregated residential load using smart meter data. *IEEE Trans. Power Syst.* 33 (5), 5446–5455. doi:10.1109/TPWRS.2018.2799903
- Quilumba, F. L., Lee, W. J., Huang, H., Wang, D. Y., and Szabados, R. L. (2015). Using smart meter data to improve the accuracy of intraday load forecasting considering customer behavior similarities. *IEEE Trans. Smart Grid* 6 (2), 911–918. doi:10.1109/TSG.2014.2364233
- Sajjad, I. A., Chicco, G., and Napoli, R. (2016). Definitions of demand flexibility for aggregate residential loads. *IEEE Trans. Smart Grid* 7 (6), 2633–2643. doi:10.1109/TSG.2016.2522961
- Shi, H., Xu, M., and Li, R. (2018). Deep learning for household load forecasting-A novel pooling deep RNN. *IEEE Trans. Smart Grid* 9 (5), 5271–5280. doi:10.1109/TSG.2017.2686012
- Stephen, B., Tang, X., Harvey, P. R., Galloway, S., and Jennett, K. I. (2017). Incorporating practice theory in sub-profile models for short term aggregated residential load forecasting. *IEEE Trans. Smart Grid* 8 (4), 1591–1598. doi:10.1109/TSG.2015.2493205
- Tolosana, R., Vera-Rodriguez, R., Fierrez, J., and Ortega-Garcia, J. (2018). Exploring recurrent neural networks for on-line handwritten signature biometrics. *IEEE Access* 6, 5128–5138. doi:10.1109/ACCESS.2018.2793966
- Vossen, J., Feron, B., and Monti, A. (2018). “Probabilistic forecasting of household electrical load using artificial neural networks,” in Proceedings of the 2018 IEEE International Conference on Probabilistic Methods Applied to Power Systems (PMAPS), 24–28 June 2018 (Boise, ID, USA: IEEE). doi:10.1109/PMAPS.2018.8440559
- Wang, L., Mao, S., Wilamowski, B. M., and Nelms, R. M. (2020). Ensemble learning for load forecasting. *IEEE Trans. Green Commun. Netw.* 4 (2), 616–628. doi:10.1109/TGCN.2020.2987304
- Wang, Y., Chen, Q., Sun, M., Kang, C., and Xia, Q. (2018). An ensemble forecasting method for the aggregated load with subprofiles. *IEEE Trans. Smart Grid* 9 (4), 3906–3908. doi:10.1109/TSG.2018.2807985
- Welikala, S., Dinesh, C., Ekanayake, M., Godaliyadda, B., Ekanayake, R. I., and Ekanayake, J. (2019). Incorporating appliance usage patterns for non-intrusive load monitoring and load forecasting. *IEEE Trans. Smart Grid* 10 (1), 448–461. doi:10.1109/TSG.2017.2743760
- Wijaya, T. K., Vasirani, M., Humeau, S., and Aberer, K. (2015). “Cluster-based aggregate forecasting for residential electricity demand using smart meter data,” in Proceedings of the 2015 IEEE International Conference on Big Data (Big Data), 29 October - 1 November 2015 (Santa Clara, CA, USA: IEEE). doi:10.1109/BigData.2015.7363836
- Wu, D., Wang, B., Precup, D., and Boulet, B. (2020). Multiple kernel learning-based transfer regression for electric load forecasting. *IEEE Trans. Smart Grid* 11 (2), 1183–1192. doi:10.1109/TSG.2019.2933413
- Xie, G., Chen, X., and Weng, Y. (2018). An integrated Gaussian process modeling framework for residential load prediction. *IEEE Trans. Power Syst.* 33 (6), 7238–7248. doi:10.1109/TPWRS.2018.2851929
- Xu, J., Meng Yue, M., Katramatos, D., and Yoo, S. (2016). “Spatial-temporal load forecasting using AMI data,” in Proceedings of the 2016 IEEE International Conference on Smart Grid Communications (SmartGridComm), 6–9 November 2016 (Sydney, NSW, Australia: IEEE). doi:10.1109/SmartGridComm.2016.7778829
- Zheng, J., Chen, X., Yu, K., Gan, L., Wang, Y., and Wang, K. (2018). “Short-term power load forecasting of residential community based on GRU neural network,” in Proceedings of the 2018 International Conference on Power System Technology (POWERCON), 6–8 November 2018 (China: GuangzhouIEEE). doi:10.1109/POWERCON.2018.8601718
- Zhou, D., Balandat, M., and Tomlin, C. (2016). “A Bayesian perspective on residential demand response using smart meter data,” in Proceedings of the 2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton), 27–30 September 2016 (Monticello, IL, USA: IEEE). doi:10.1109/ALLERTON.2016.7852373
- Zou, M., Fang, D., Harrison, G., and Djokic, S. (2019). “Weather based day-ahead and week-ahead load forecasting using deep recurrent neural network,” in Proceedings of the 2019 IEEE 5th International forum on Research and Technology for Society and Industry (RTSI), 9–12 September 2019 (Florence, Italy: IEEE). doi:10.1109/RTSI.2019.8895580