

Research Article

Unified Quantile Regression Deep Neural Network with Time-Cognition for Probabilistic Residential Load Forecasting

Zhuofu Deng ¹, Binbin Wang ^{1,2}, Heng Guo ¹, Chengwei Chai ¹, Yanze Wang ³,
and Zhiliang Zhu ¹

¹College of Software, Northeastern University, Shenyang 110169, China

²Deep NEU Technology Co., Ltd., Shenyang 110169, China

³Kuandian Electric Power Supply Company, State Grid Liaoning Electric Power Supply Co., Ltd.,
Kuandian 118200, Liaoning, China

Correspondence should be addressed to Binbin Wang; neubob414@163.com

Received 22 July 2019; Revised 6 December 2019; Accepted 21 December 2019; Published 22 January 2020

Academic Editor: Ning Cai

Copyright © 2020 Zhuofu Deng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Residential load forecasting is important for many entities in the electricity market, but the load profile of single residence shows more volatilities and uncertainties. Due to the difficulty in producing reliable point forecasts, probabilistic load forecasting becomes more popular as a result of catching the volatility and uncertainty by intervals, density, or quantiles. In this paper, we propose a unified quantile regression deep neural network with time-cognition for tackling this challenging issue. At first, a convolutional neural network with multiscale convolution is devised for extracting more behavioral features from the historical load sequence. In addition, a novel periodical coding method marks the model to enhance its ability of capturing regular load pattern. Then, features generated from both subnetworks are fused and fed into the forecasting model with an end-to-end manner. Besides, a globally differentiable quantile loss function constrains the whole network for training. At last, forecasts of multiple quantiles are directly generated in one shot. With ablation experiments, the proposed model achieved the best results in the AQS, AACE, and inversion error, and especially the average of the AACE is grown by 34.71%, 75.22%, and 32.44% compared with QGBRT, QCNN, and QLSTM, respectively, indicating that our method has excellent reliability and robustness rather than the state-of-the-art models obviously. Meanwhile, great performances of efficient time response demonstrate that our proposed work has promising prospects in practical applications.

1. Introduction

The power system is one of the most complex man-made systems, and electric load forecasting plays a vital role in power system planning and operations, revenue projection, rate design, electricity market trading, and so forth. Electric load forecasting categories can be simply summarized as follows: very short-term load forecasting (VSTLF), short-term load forecasting (STLF), medium-term load forecasting (MTLF), and long-term load forecasting (LTLF). STLF studies data with hourly temporal resolution and has forecasting horizon from hours to days [1]. STLF gives great significances to power systems in providing strategies,

reliability analysis, interchange evaluation, security assessment, and spot price calculation [2].

Recently, a mass of smart meters has been installed around the globe, producing plenty of fine-grained electric consumption data from time to space. High-resolution data from smart meters provide a wealth of information about consumers' power consumption behaviors and lifestyles, creating opportunities for accurate residential load forecasting. Residential load forecasting is becoming more and more important for many entities in the electricity market. For consumers, residential load forecasts can be used as the input of home energy management systems (HEMS) to decrease the cost and expense [3]. In the future electricity

market, it also has potentials in the peer-to-peer (P2P) market [4, 5]. For retailers, residential load forecasting serves for pricing, purchasing, and hedging decisions and maximizes retailers' profits [6]. To aggregators, it is utilized to produce more accurate aggregate load forecasts by clustering or other methods [7, 8]. In distribution system operators (DSO), depending on effective residential load forecasting, peak load reduction can be achieved by flexible use of the energy storage (ES) system or intelligent demand response (DR) technology. Accurate load forecasting for residential customers is also able to help DSO to locate the best customer groups who most likely participate in demand response planning, which reflects great significances for supplying load-balancing reserves and hedging market costs.

Traditional load forecasting concentrates on various aggregation levels, such as system, feeder, and regional level. Loads of these levels filter out many random fluctuations; hence, the profile is smoother and more regular. On the contrary, due to the difference of customers' lifestyle and the randomness of behavior, the load profile of single residence shows more volatility and uncertainty and the profile characteristics of different customers also provide great diversity.

Normalized hourly profiles of three different level loads are shown in Figure 1, illustrating that residential load profile presents more volatility compared with higher-level load profile. Therefore, residential load forecasting is more challenging, and models that perform well at the aggregate level are no longer suitable for residential level.

Most existing literatures focus on point forecasting techniques, attempting to forecast the expected value of future load. However, residential load is far more volatile and uncertain than aggregate level; therefore, highly reliable point forecasts are difficult to produce. As a result, more and more decision processes begin to rely on probabilistic load forecasting (PLF), which contains more probabilistic information and generates output in the form of intervals, density, or quantiles.

This paper proposed a unified quantile regression deep neural network with time-cognition to produce probabilistic residential load forecasting. At first, a deep neural network with multiscale convolution is proposed for extracting more discriminative features from the historical load sequence. In addition, a periodical coding method is devised to mark the model for capturing regular periodical load pattern. Besides, the outputs of both branches are fused and inputted into the proposed model for residential load forecasting with an end-to-end manner. Meanwhile, we introduce a globally differentiable quantile loss function to constrain the whole network in the training process. At last, forecasts of multiple quantiles are directly generated in one shoot at the end of the neural network. The main contributions of the paper are summarized as follows:

- (i) A unified quantile regression deep neural network with time-cognition is proposed for tackling the probabilistic residential load forecasting problem
- (ii) Comprehensive and extensive experiments are conducted for inspecting reliability, sharpness, robustness, and efficiency of the proposed method

- (iii) We introduce the quantile inversion error as a complementary metric to detect the robustness of the quantile regression model

To the best of our knowledge, this is the first paper that presents the quantile loss supervised CNN-based model with time-cognition for probabilistic residential load forecasting. In addition, the quantile inversion error is adopted to verify the robustness particularly.

This paper will be structured as follows: Section 2 provides the background of the load forecasting community. Section 3 defines the problem and describes the details of our proposed method. Section 4 introduces the experiment setups. Section 5 reports the experimental results and further discusses some insights of the proposed method. The conclusions are drawn in Section 6.

2. Related Work

Load forecasting approaches have been extensively and intensively studied. The present works mostly pay more attentions on different aggregation levels for better forecasting, such as system, feeder, and regional level. This strategy could remove uncertainties present in residential load behaviors as far as possible so that the prediction for clusters is facilitated with a more regular pattern. In contrast, for the issue of forecasting to individual residences, conventional models inevitably suffer from customers' lifestyle and the randomness of behaviors resulting in providing poor performances in load forecasting.

Some research studies have examined popular models specialized in the aggregation level for residential load forecasting. Humeau et al. [9] investigated the application of linear regression (LR), multilayer perceptron (MLP), and support-vector machine for regression (SVR) in residential load forecasting and found that LR has higher accuracy in single residential level, while SVR performs better in aggregation level. In the case of applying a specific clustering algorithm, SVR outperforms LR when the number of the cluster is greater than 32 houses. Edwards et al. [10] evaluated familiar machine learning algorithms focusing on building load forecasting to residence, including LR, feed-forward neural networks (FFNN), SVR, and their variants. Experiments showed that they are generally uncomfortable on residential data, but least-squares support-vector machines (LS-SVM) performs well. Ahmadihangar et al. [11] imported the generalized linear mixed-effects (GLME) model to generate load patterns for forecasting the potential flexibility of residential customers. The advantage of this method is that it can be used, in on-line and real-time methods, in a wide range of control approaches.

Recent research studies are no longer limited to traditional regression or time-series methods, but rather follow designing more sophisticated models to capture advanced features related to residential behaviors. Tascikaraoglu and Sanandaji [12] proposed a new forecasting approach that combines compressive sensing (CS) and data decomposition, providing a framework which facilitates exploiting the existing low-dimensional structures governing the interactions among

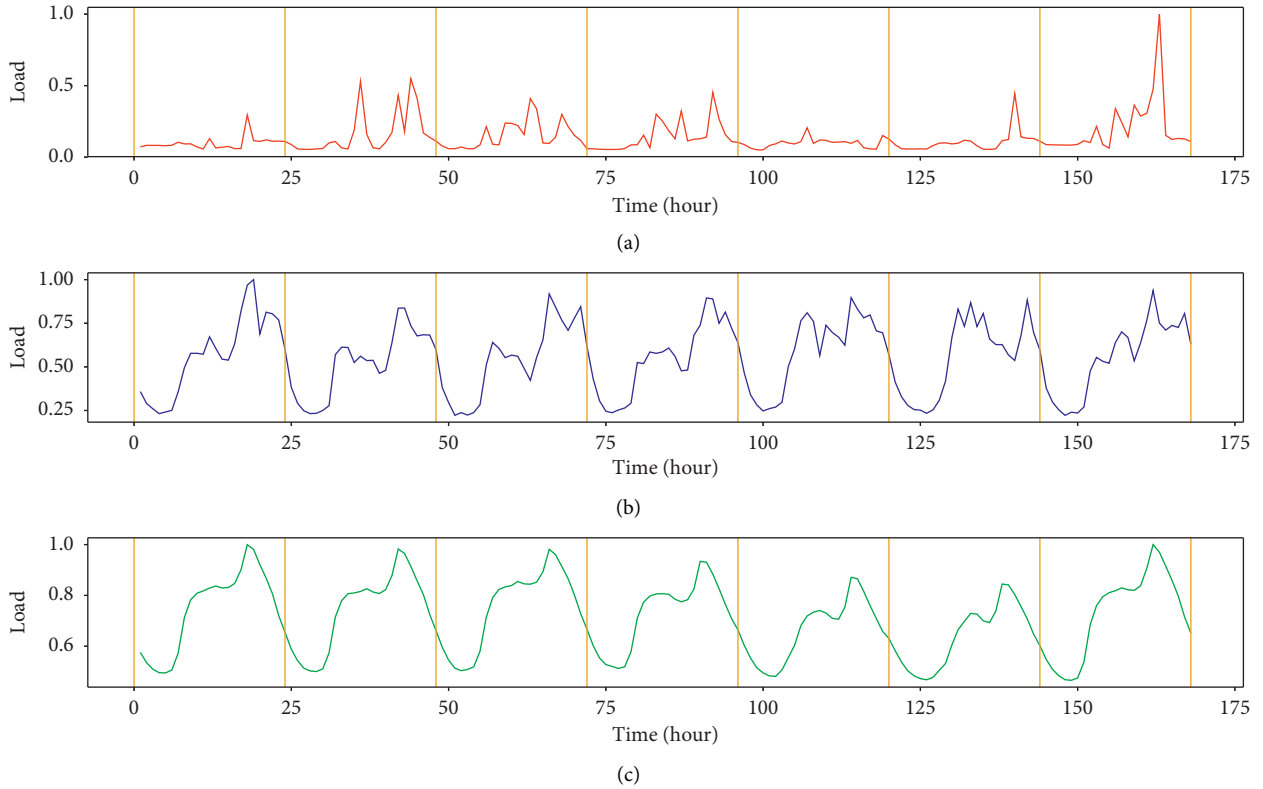


FIGURE 1: Three normalized hourly profiles of different level loads including (a) residential level, (b) feeder level, and (c) system level.

residences. Yu et al. [13] utilized sparse coding features in load forecasting for individual households and provided a large-scale evaluation on the proposed method with several classical models. Experiments showed that sparse coding features are efficient in decreasing the forecasting error of next-day and next-week total load. Similarly, Pan et al. [14] also relied on the sparse characteristics in residential loads. A method based on least absolute shrinkage and selection (LASSO) is proposed to adaptively explore sparsity in historical data and leverage predictive relationship among different residences, and its low computational complexity and high accuracy are verified by experiments. Recently, deep learning (DL) has become a research hotspot of artificial intelligence applications in many fields due to its powerful feature extraction and fitting capabilities [15]. Abbas et al. [16] proposed a unique and improved nonlinear autoregressive neural network with external input- (NARXNN-) based recurrent load forecaster using a lighting search algorithm (LSA). Experiment was conducted on substation-level aggregated residential load data, achieving 16%–20% improvement of precision in comparison with present computational techniques. Long short-term memory (LSTM) and gated recurrent unit (GRU), variants of the recurrent neural network (RNN), are introduced to resolve the individual residential load forecasting problem in [17–19], reflecting remarkable predominance in forecasting accuracy over traditional machine learning algorithms and fully connected neural networks.

Great literatures focus on point forecasting techniques, attempting to predict the possible consumption in future.

However, residential load is far more volatile and uncertain than aggregate level, providing difficulties to get high reliable point forecasts. Consequently, more and more algorithms of load forecasting prefer probabilistic load forecasting that provides probabilistic results in the form of intervals, density, or quantiles. Probabilistic load forecasting methods can be divided into three categories according to different generation processes:

- (i) Multiple scenarios are devised and fed into point forecasting models, and then outputs of these point forecasts are generated to form a probabilistic prediction. Many different temperature scenario generation models have been proposed, including fixed-date [20], shifted-date, bootstrap [21], and surrogate models [22]. Xie and Tao [23] compared these methods and revealed that the shifted-date model acquired superior performance when the number of dates being shifted locates within a reasonable range.
- (ii) Modeling techniques are applied in probabilistic load forecasting directly, such as density estimation [24, 25], Gaussian process regression [26], and quantile regression [27]. The original probabilistic forecasting techniques have not won more attentions from the load forecasting community over the past thirty years. Instead, some of these techniques have been used for generating point load forecasts, which are essentially the expected values derived

from the probabilistic forecasts. One possible explanation for the underdevelopment of these probabilistic forecasting techniques for load forecasting is the fact that their point forecast accuracy may not be as good as those from point forecasting techniques. Before the establishment of formal evaluation methods for PLF, people may have underestimated the power of these probabilistic forecasting techniques based on their underperformance in point forecasting accuracy.

- (iii) Postprocessing of the point forecasting model is able to produce probabilistic results by estimating density function's parameters of residual or combining several outputs of multiple point forecasting models. Xie et al. [28] evaluated several residual simulation methods, demonstrating that residuals cannot always obey normal distribution and proposed several skills to tackle this problem for optimizing performances. Liu et al. [29] proposed a hybrid method to generate probabilistic load forecasts by performing quantile regression averaging on sister point forecasts.

Due to great volatility and uncertainty, probabilistic forecasting load for residences is extremely challenging and only a few literatures have made some preliminary attempts. Shepero et al. [30] introduced the log-normal process (LP) model that is designed for positive data like residential loads with combination of log-normal distribution and Gaussian processes. Traditional Gaussian processes and the log-normal process were evaluated in ablation studies and it was found that the log-normal process produces sharper probabilistic forecasts. Gerossier et al. [31] proposed a quantile smoothing spline regression and used three inputs: hourly load in previous day, median load of the previous week, and temperature prediction. The robustness of this approach is enhanced by fallback models to overcome defective data such as insufficient data, unavailable variables, and extreme situations. Experiments reflected that this model consistently outperforms the persistence model and provides more reliable probabilistic forecasts. Ben Taieb et al. [32] proposed an additive quantile regression model for a set of quantiles of the future distribution using a boosting procedure, which includes flexible and interpretable models with an automatic variable selection. The authors confirmed the advancement of their proposed approach with three benchmark methods on both aggregated and disaggregated scales using a smart meter dataset.

3. Methodology

In this section, we further analyze the characteristics of residential load and study how to utilize deep learning technology to achieve more advanced performance. Finally, our proposed model is introduced in detail.

3.1. Problem Identification. Aggregate load forecasting integrates multiple load subprofile to filter out uncertainty and volatility, which enables the load curve pattern smoother

and more regular for superior prediction. However, residential load consumption lacks a stable and consistent pattern compared with aggregate-level loads. In some scenarios, people's behaviors are more regular, such as getting up in the morning, which is described as "general understanding" [8]. In some cases, behaviors are irregular even disordered. For example, the lifestyle of individuals or families with appliances results in inhomogeneous patterns in households. For example, some people prefer more social activities after work; thus, their contributions to load consumption in the evening are relatively limited, but people who stay indoors may give another condition. In fact, the potential load pattern for every household may be different that increases difficulties for load forecasting to residences. Figure 2 draws the daily load profiles of three residences. Figure 2(a) shows a residence with regular lifestyle. The profiles corresponding to Figure 2(b) is irregular; hence, the prediction is more challenging and dependent on the instant law. The behavior in Figure 2(c) grows more regular in the morning, while in other terms, there are no fixed-time activities.

Therefore, an advanced model for residential load forecasting should be able to capture the characteristics of activities in history. Meanwhile, it should also extract the law of electricity consumption in narrow span related to the forecasting point. If the model combines both features in a reasonable way, this method would hold great generalization to different load patterns of households and achieve remarkable performance. Traditional regression models, such as SVR [33, 34] and Gradient Boost Regression Tree (GBRT) [35, 36], are not designed for time-series problems. Other time-series models, such as AR and ARIMA [37, 38], can capture features in historical load sequences, but cannot effectually combine them with instant behavioral characteristics to produce better results.

Deep neural networks are one of the most promising technologies available today. Due to the excellent capability of learning discriminative features, the network can explore potential characteristics of historical load sequences efficiently. At the same time, the model can also perform regression learning on temporal features. Therefore, the deep neural network can naturally integrate two learned representations and finally realize learning complex rules in an end-to-end approach. Among typical architectures of deep learning, recurrent neural networks (RNNs) and convolutional neural networks (CNNs) are proper candidates suited for tackling this issue. LSTM [39, 40] is good at solving the problem of gradient disappearance/explosion due to excessive sequence length, allowing for longer information to be remembered. LSTM has been successfully used to resolve load forecasting problem in [17, 18, 41]. However, LSTM still has two troubles: the efficiency of recursive operations and the limited memory capacity for dealing with the longer time-series problem.

CNN has made revolutionary progress in computer vision, and it has been widely studied in other fields in recent times. Recent works demonstrate that neural networks with convolution operations can achieve top performance in sequence tasks such as speech synthesis, language modeling,

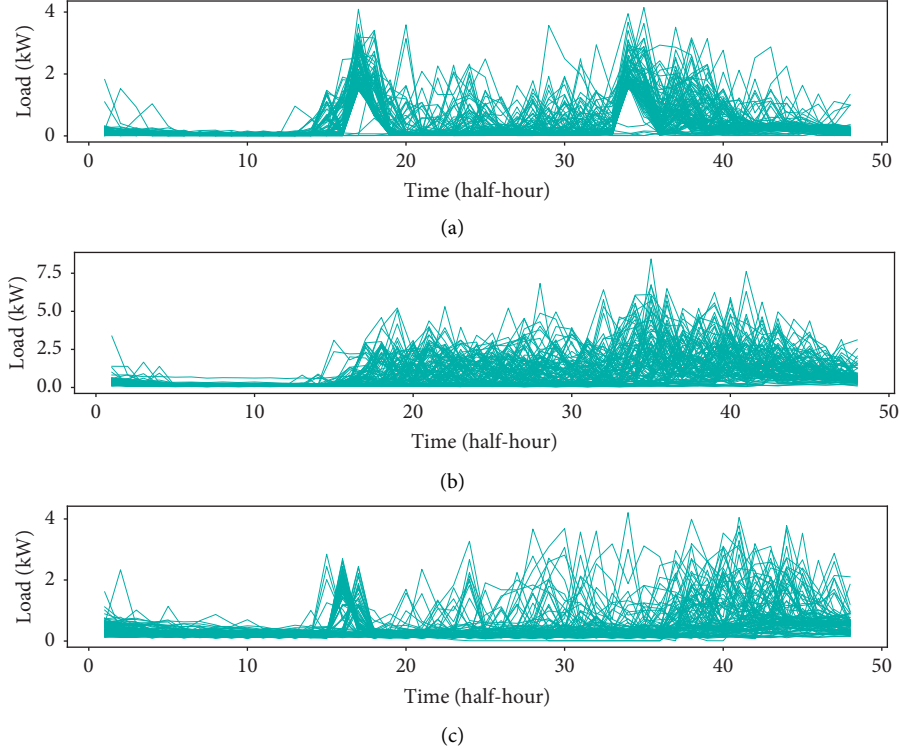


FIGURE 2: Different residential load patterns. (a) Meter ID: 5843. (b) Meter ID: 6902. (c) Meter ID: 7374.

and machine translation [42–44]. CNN-based networks are also used for load forecasting problems in [45–47]. Based on the analysis of residential load forecasting, we propose a unified quantile regression deep neural network with time-cognition, which consists of sequence-to-sequence (S2S) multi-scale CNN structure (MS-CNN), periodic time coding, and quantile regression components.

3.2. S2S MS-CNN. Ordinary one-dimensional CNNs can only catch relationships in neighborhood elements, which limit the capabilities of extracting discriminative features for detecting potential rules efficiently. Therefore, multiscale convolution is introduced to fuse extra nonadjacent ones in enlarged receptive fields. Specifically, deep neural network stacks multiple dilated convolutional layers with different scales [48, 49]. Multiscale convolution extracts rich and crucial relationships from different input locations, but prevents great growth of trainable parameters as well.

Traditional one-dimensional CNNs can accept very long input sequences, but only output one forecast, which limits its training efficiency and gradient flowing. Inspired by RNNs, the proposed model uses a S2S structure in which each point in the input sequence corresponds to a forecast. This structure enhances the flow of gradients, forcing the information at each moment to be effectively extracted into the model, thereby improving the efficiency and accuracy of the training. In order to avoid future functions, we use a causal convolution to ensure that any node only acquires information about the past. Figure 3 shows a schematic diagram of a block containing four multiscale one-dimensional causal convolution layers with the kernel size of 2. The

dilation rate s increases along l -th layers within a block such that $s_l = 2^l$.

The proposed structure consists of a stack of blocks, each of which contains L convolutional layers. The output of each layer is connected by a residual connection, that is, the input of each layer is added to its output. Let T be the length of input sequence and F_w be the number of filters, then $H^{(j,l)} \in \mathbb{R}^{F_w \times T}$ presents the convolution of the l -th layer of j -th block, noting that T and F_w keep consistent in all layers so that output from different layers can be added. Meanwhile, except for convolutional operation, a layer also holds a series of operations $g(\cdot)$ including activation, normalization, and dropout. Taking a convolution filter size of 2 as an example, the convolution operation is applied at positions $t-s$ and t . The filters' parameters are denoted as $W = \{W^{(1)}, W^{(2)}, b\}$, where parameter $W^{(i)} \in \mathbb{R}^{F_w \times F_w}$ and bias $b \in \mathbb{R}^{F_w}$. Let $\hat{H}_t^{(j,l)}$ be the layer's output at time t and $H_t^{(j,l)}$ be output of the residual connection such that

$$\begin{aligned} \hat{H}_t^{(j,l)} &= g(W^{(1)}H_{t-s}^{(j,l-1)} + W^{(2)}H_t^{(j,l-1)} + b), \\ H_t^{(j,l)} &= H_t^{(j,l-1)} + U\hat{H}_t^{(j,l)} + e, \end{aligned} \quad (1)$$

where $U \in \mathbb{R}^{F_w \times F_w}$ and $e \in \mathbb{R}^{F_w}$ are the weights and biases of the residual convolution with a filter size of 1. The proposed structure is shown in Figure 4, which consists of 8 blocks, and each block holds 4 dilated convolution layers.

3.3. Time-Cognition. Time is a key feature of forecasting problems, and most current models adopt the one-hot encoding approach [41, 50], which has two main challenges.

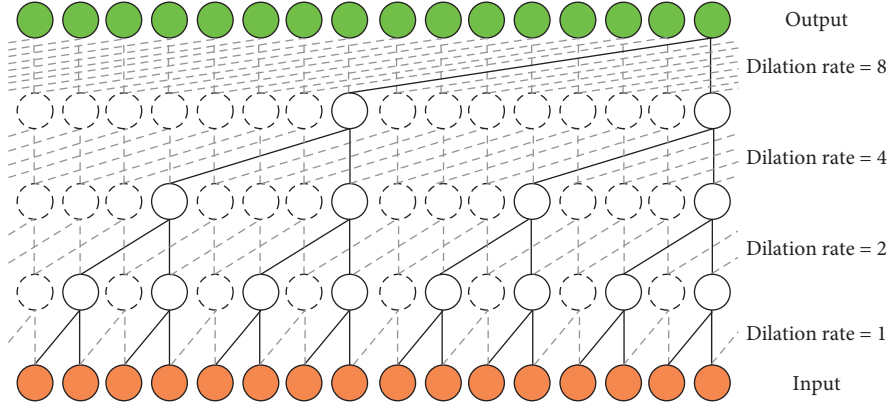


FIGURE 3: A simple instance of a multiscale one-dimensional causal convolution block. This block includes four multiscale convolutional layers with the filter size of 2, resulting in a receptive field size of 16 for a top-layer node. “Causal” means that each hidden node in this network only obtains information about the past. The dilation rate is 1, 2, 4, and 8 from the bottom to top. The first convolutional layer with dilation rate 1 is an ordinary causal convolutional layer, and other layers with dilation rate greater than 1 could fuse features of multiple time scales.

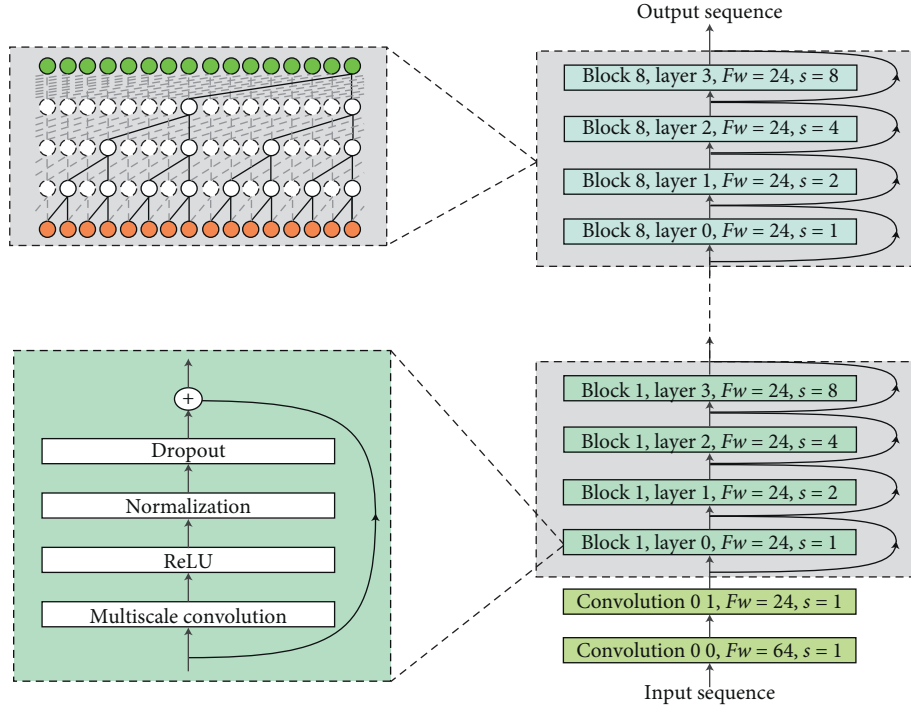


FIGURE 4: The main structure of the MS-CNN is on the right side. The left-top part shows dilated convolution of a block, and the left-bottom part describes detailed structure in one layer.

First, the one-hot encoding approach does not take into account the periodic relationships between points in time, which makes the neural network lose prior knowledge resulting in a precision to be improved. Second, the length of the one-hot vector depends on the number of points in one period. For example, encoding days of a year will make the length of the vector as long as 366, which not only increases the calculation but also reduces the forecasting accuracy. The approach provides a unique coding to each moment in one period and reserves all periodic characteristics of time.

Specifically, periodic coding $[p_t^{\sin}, p_t^{\cos}]$ corresponding to moment t can be generated by the sine and cosine functions:

$$\begin{cases} p_t^{\sin} = \sin \frac{2\pi t}{C}, \\ p_t^{\cos} = \cos \frac{2\pi t}{C}, \end{cases} \quad (2)$$

where C describes the length of a period. In this paper, we adopt the half-hours in one day, days in one week, and days

in one year as the time feature and then code them to $[p^{hd \sin}, p^{hd \cos}]$, $[p^{dw \sin}, p^{dw \cos}]$, and $[p^{dy \sin}, p^{dy \cos}]$, respectively. At last, we obtain the full coding of time t : $p_t = [p_t^{hd \sin}, p_t^{hd \cos}, p_t^{dw \sin}, p_t^{dw \cos}, p_t^{dy \sin}, p_t^{dy \cos}]$. Different coding methods are compared in Figure 5.

3.4. Quantile Loss. The proposed CNN structure can generate point forecasts or probabilistic forecasts by applying different supervised techniques. Point regression methods model the average behavior, which is useful but gives less information of the forecasts. Quantile regression allows us to get the forecasts at different quantile levels, hence drawing a more comprehensive picture of the forecasted moment. Quantile regression not only makes it easy to get multiple quantile forecasts but also allows calculating the prediction interval (PI). The quantile loss can be written as

$$L_{t,q} = (y_t - \hat{y}_{t,q})(q - \mathbb{1}\{y_t < \hat{y}_{t,q}\}), \quad (3)$$

where y_t is the truth at time t and $\hat{y}_{t,q}$ denotes the forecast of quantile q at time t .

In the process of training, the back-propagation algorithm requires the loss function to be differentiable so that the network can be trained using gradient descent. Common pinball loss is not differentiable everywhere; therefore, we introduced the log-cosh function to approximate the quantile loss function, with least change, making the loss function differentiable everywhere [51]. Then, we obtain the new loss function as follows:

$$L_{t,q} = \begin{cases} \frac{\log(\cosh(q\gamma|y_t - \hat{y}_{t,q}|))}{\gamma}, & y_t - \hat{y}_{t,q} \geq 0, \\ \frac{\log(\cosh((1-q)\gamma|y_t - \hat{y}_{t,q}|))}{\gamma}, & y_t - \hat{y}_{t,q} < 0, \end{cases} \quad (4)$$

where γ is a hyperparameter that tunes the bound between L1 norm and L2 norm and should be chosen according to the value size of data and the expected accuracy. Higher accuracy will call for a bigger γ . $\log(\cosh(x))$ is approximately equal to $x^2/2$ when x keeps very small, and it tends to be approximately equal to $|x| - \log(2)$ when x becomes larger.

3.5. Proposed Network Architecture. Previously mentioned techniques are integrated into the entire network as key components, the details of which are shown in Figure 6. At first, a historical load sequence is inputted into the MS-CNN network and representation vectors are generated from the last layer of the network, which contains discriminative features for load pattern of the historical data in moment. Then, an auxiliary fully connected network learns the periodic codes of the forecasted time to increase extra features for prediction. At last, the output vectors of two subnetworks are fused and fed to another fully connected layer to produce all quantile forecasts. The proposed model reflects the contribution of the multiscale convolution neural network

| Days of the week | Nature coding | One-hot coding |
|------------------|---------------|-----------------|
| Sunday | 0 | [1 0 0 0 0 0] |
| Monday | 1 | [0 1 0 0 0 0] |
| Tuesday | 2 | [0 0 1 0 0 0] |
| Wednesday | 3 | [0 0 0 1 0 0] |
| Thursday | 4 | [0 0 0 0 1 0] |
| Friday | 5 | [0 0 0 0 0 1] |
| Saturday | 6 | |
| | | Periodic coding |
| | | [0.00 1.00] |
| | | [0.78 0.62] |
| | | [0.98 -0.22] |
| | | [0.43 -0.90] |
| | | [-0.43 -0.90] |
| | | [-0.98 -0.22] |
| | | [-0.78 0.62] |

FIGURE 5: Coding method comparison by taking seven days of one week as an example.

and periodic coding especially with feature fusion or concatenation mechanism.

Individual electricity consumption behavior naturally exhibits a time-series curve. To highlight the idea of our proposed approach, only historical load data and calendar data in the form of periodic coding are utilized as input features in this paper. It is worth noting that other environmental factors, such as temperature, can also be employed in this model. Let \mathbf{X} be the entire historical load data with length T , which is divided into input load sequences $X_t = [x_{t-m+1}, x_{t-m+2}, \dots, x_{t-1}, x_t]$ with length m and start moment $t-m+1$. X_t is a two-dimensional array with shape $(m, 1)$. It is worth noting that when T is small, too large m will make the convergence speed extremely slow like the batch size. The auxiliary input corresponding to time t is the periodic coding of next moment $p_{t+1} = [p^{hd \sin}, p^{hd \cos}, p^{dw \sin}, p^{dw \cos}, p^{dy \sin}, p^{dy \cos}]$, which is a vector with length 6. The output about time t includes quantile forecasts of time $t+1$: $\hat{Y}_{t+1} = [\hat{y}_{t+1,1}, \hat{y}_{t+1,2}, \dots, \hat{y}_{t+1,Q}]$, where Q is the number of quantile.

The implementation process is divided into three stages: data preparation, model training, and forecasting, and the details are described in Figure 7. We trained the proposed model for each customer with shared parameters for households. During the training, we used the learning rate decay and early stopping strategies based on the variation of validation loss to reduce computation cost and prevent overfitting.

4. Experiment Setup

This section introduces the experiment setups, including the data description, training implementation, and platforms.

4.1. Data Description. The dataset adopted in this paper comes from the customer behavior trials (CBT), which is a smart metering project launched by the Commission for Energy Regulation, the regulator for the electricity, and natural gas sectors in Ireland. It includes thousands of residential customers' anonymous data, which sample half-

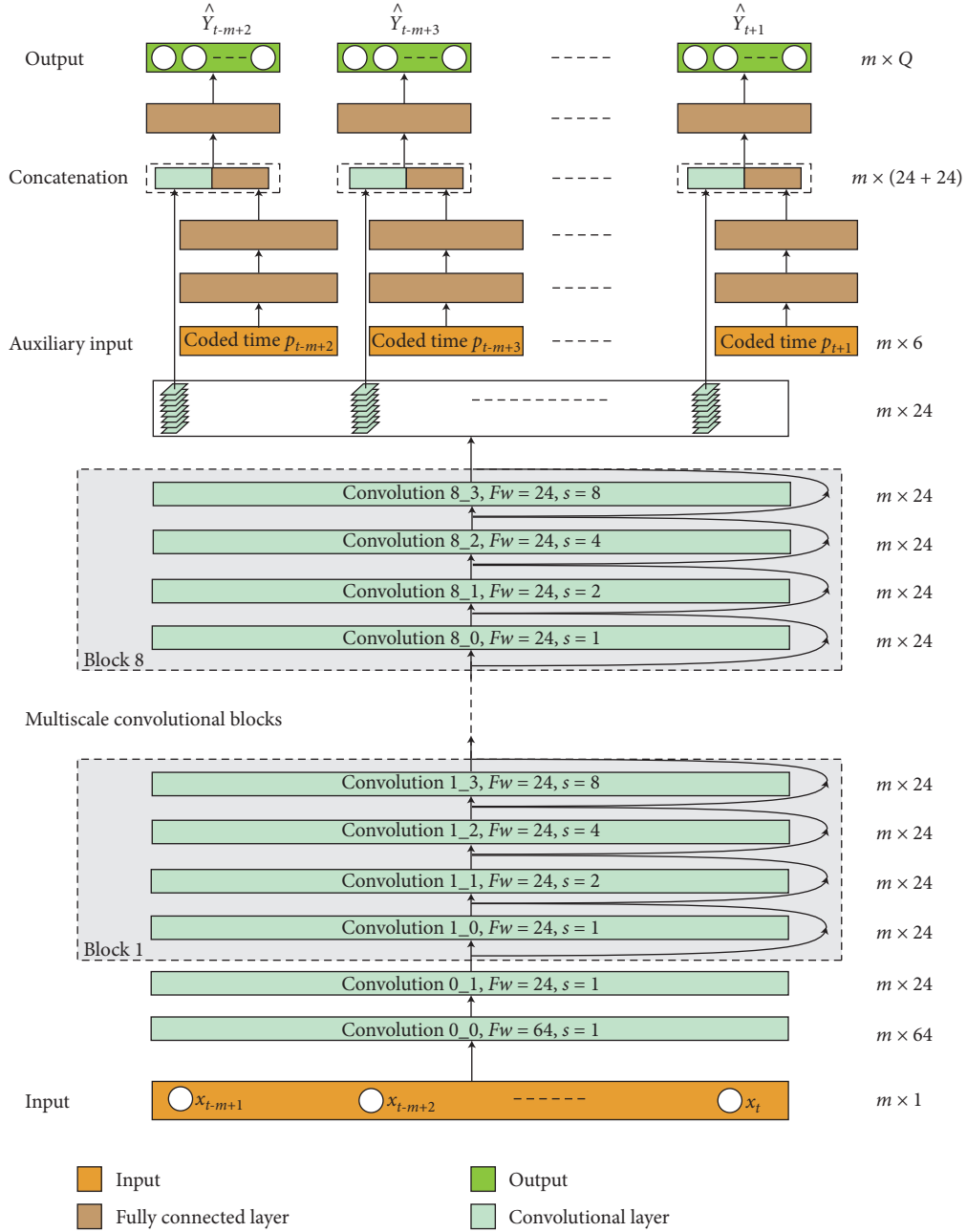


FIGURE 6: Two subnetworks MS-CNN and fully connected networks build up the baseline of proposed model for multiscale and periodical coding features fusion.

hourly electric consumption from each residence. The dataset is collected from July 15, 2009, to January 1, 2011, and each customer's data contains 25,728 collection points.

Because the dataset filled with lots of noises, we removed some redundant data and filled the missing data by the linear interpolation algorithm. Moreover, 20 residential load profiles are randomly selected from the dataset as subjects for algorithm evaluation. Data of each subject are divided into training set, validation set, and test set in order at 80%, 10%, and 10% respectively.

4.2. Benchmarks and Setup. Quantile regression LSTM (QLSTM) and quantile gradient boosting regression tree (QGBRT) are selected as experimental benchmarks. Among the traditional method, the QGBRT is often adopted for its stable and accurate performance. The LSTM has achieved top performance in many sequence-processing areas. The QLSTM model proposed in [41] has also achieved state-of-the-art performance for probabilistic load forecasting. To observe the improvement of our proposed model, a common sequence-to-point quantile CNN (QCNN) was also

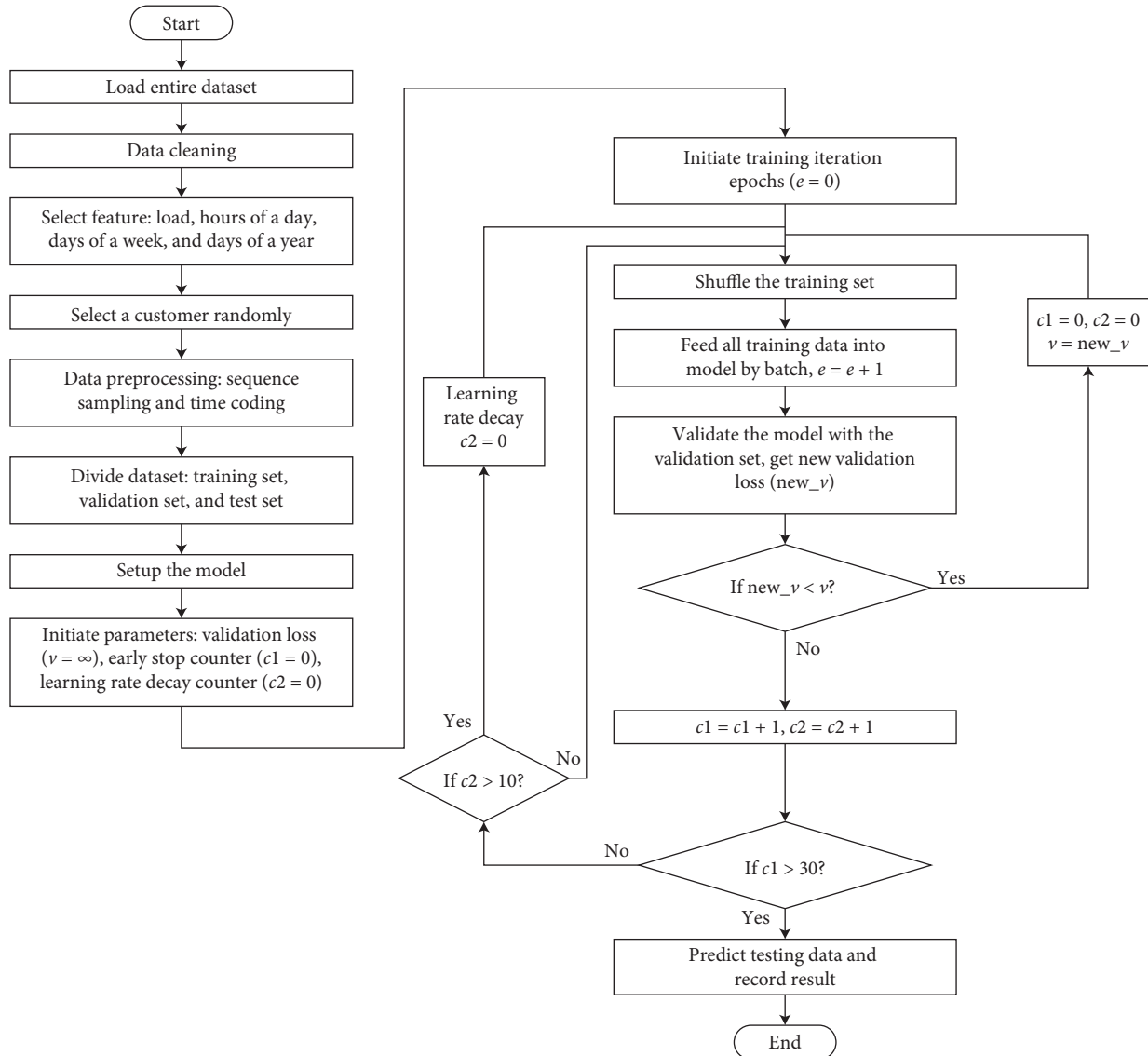


FIGURE 7: Flowchart for implementation of the proposed model.

introduced as a benchmark to verify the improvement of our proposed model.

Hyperparameter tuning has close relationship with the forecasting performance. It is not advisable to explore the best parameters for each subject, which definitely results in great computation cost. In our work, we paid more attentions on proper overall performances of all subjects. Therefore, some rules of thumb for hyperparameter selection are adopted. In general, deeper and wider network tends to achieve better performance, and for proper evaluation, we kept the scale of trainable parameters similar with the QLSTM model. The input size of QCNN, QLSTM, and our proposed model is the same to ensure the fairness, which is set to 240. If the length of the input sequence cannot extract sufficient features for growing ability of extracting discriminative features and if too long, the efficiency of models in computation is limited and sometimes causes overfitting problem. The QCNN and the proposed model share some

hyperparameters: 34 convolutional layers and each convolutional layer consists of a convolutional operation, a rectified linear unit (ReLU), a batch normalization filter, and one dropout action. QLSTM stacks 3 units in order to improve forecasting capacity. In addition, we employed grid search to explore other hyperparameters, such as kernel size, dropout rate, start learning rate, batch size, and so forth. In this experiment, we made probabilistic forecasts with 9 quantiles (0.1–0.9) of halfhour for evaluation on our proposed model; therefore, the output sequence has the shape $m \times Q$. The experiment setups and hyperparameters of neural networks are presented in Table 1.

4.3. Software and Hardware Platform. All experiments were executed on a cloud server with two NVIDIA P4 computing cards and the CPU with 8 cores. The implementations of QGBRT are based on the scikit-learn packages. Other neural

TABLE 1: Shared parameters of neural networks and related strategies selected.

| Parameters | LSTM | CNN |
|---------------------|----------------|----------------|
| Depth | 3 | 34 |
| Hidden neural | 128 | None |
| Kernel size | None | 8 |
| Kernel number | None | 24 |
| Batch size | 128 | 128 |
| Input size | 240 | 240 |
| Dropout rate | 0.05 | 0.05 |
| Loss function | Quantile loss | Quantile loss |
| Optimizing method | AMSGrad | AMSGrad |
| Start learning rate | 0.01 | 0.01 |
| Learning rate decay | 0.3 | 0.3 |
| Training stop | Early stopping | Early stopping |

network-based models are realized by the Keras framework with TensorFlow [52] backend.

5. Results and Discussion

This section introduces the experiment setup, including the evaluation metrics and experiment result.

5.1. Evaluation Metrics. For probabilistic forecasting evaluation, there are three commonly used attributes: reliability, sharpness, and resolution. Reliability refers to how close the predicted distribution is to the ground truth. Sharpness means how tightly the predicted distribution covers the actual curve. Resolution signifies how much the predicted interval varies over time. Measures like Kolmogorov–Smirnov, Cramer–von Mises, and Anderson–Darling statistics assess the unconditional coverage of a probabilistic forecasting rather than its sharpness or resolution. In this paper, the performance of the probabilistic forecasting is evaluated by the average quantile scores (AQS), which is a comprehensive measure metrics considering not only reliability but also sharpness and resolution. The quantile scores have the same equation with quantile loss, and AQS is defined as follows:

$$\text{AQS} = \frac{1}{T_{\text{test}} \times Q} \sum_{t=1}^{T_{\text{test}}} \sum_{q=1}^Q L_{t,q}, \quad (5)$$

where Q denotes the defined number of quantiles and T_{test} denotes the number of samples in test set. In addition, in order to make a proper evaluation on candidates, the prediction interval (PI) should be assessed. PI of time t with $100(1 - \delta)\%$ confidence level is given as $[L_t^\delta, U_t^\delta]$, where L_t^δ and U_t^δ are the lower and the upper boundaries of the PI. It is expected that the ground truth y_t should be between the lower and the upper boundaries with the nominal probability $100(1 - \delta)\%$, which is called prediction intervals' nominal confidence (PINC). The PI coverage probability (PICP) evaluates whether the actual value lies within a certain prediction interval limit:

$$\text{PICP} = \frac{1}{T_{\text{test}}} \sum_{t=1}^{T_{\text{test}}} \mathbb{1}\{y_t \in [L_t^\delta, U_t^\delta]\}, \quad (6)$$

where T_{test} is the sample number of the test set. Results with high reliability should get the PICP close to the PINC [53]. So, we use the absolute average coverage error (AACE) to evaluate the reliability:

$$\text{AACE} = |\text{PICP} - \text{PINC}|. \quad (7)$$

Another metric, PI normalized average width (PINAW) [54], is introduced to measure the sharpness of the PI. The PINAW is defined as

$$\text{PINAW} = \frac{1}{T_{\text{test}} R} \sum_{t=1}^{T_{\text{test}}} (U_t^\delta - L_t^\delta), \quad (8)$$

where R is the difference between maximum and minimum of the ground truth and normalizes the PI average width. In this paper, we evaluated the PI coverage of 80% for AACE and PINAW. Lower AQS, AACE, and PINAW indicate a better performance.

5.2. Result Analysis. We randomly selected 20 residences and trained models separately for each residence which belongs to the probabilistic forecasting model and is able to output 9 quantiles in one shot. To ensure the fairness of experiments, all subjects in training or testing phases kept consistent parameters and model architecture. Figure 8 is a heat map of AQS results from studied models for comparisons, intuitively revealing that our proposed model serves the most sophisticated performance. Specifically, our proposed model achieved 12 best results in 20 residences and the rest also performed excellently. QGBRT contained 6 best results outperforming QLSTM and QCNN. The right side of Figure 8 reflects AQS improvement ratio of our proposed model over other three algorithms. Statistically, our model acquired the most excellent results with an average growth of 2.23%, 7.15%, and 2.12% compared with QGBRT, QCNN, and QLSTM, respectively. Meanwhile, the classic machine learning model QGBRT gave promising evaluations after proper adjustments. Since the crude one-dimensional CNN model lacks effective optimization for sequence problems, its performance is restricted to weak ability of learning discriminative features with simple convolutions.

Table 2 lists results of AQS in perspectives of each quantile. I_QGBRT, I_QCNN, and I_QLSTM represent the AQS improvement ratio comparing our proposed model with other three models. AQS values of quantiles 0.1 and 0.2 are relatively lower since the quantity of household electricity consumption is maintained in a low level with nonnegative character. In comparison of four models, our proposed model achieved the best performance in all quantiles and averages, indicating that our model has an excellent stability crossing quantiles.

Heat map of 80% interval AACE is described in Figure 9. Our proposed model achieved 11 best results of 20 residences, and QLSTM, QGBRT, and QCNN held 5, 3, and 1 ones, respectively. In terms of overall performance, most

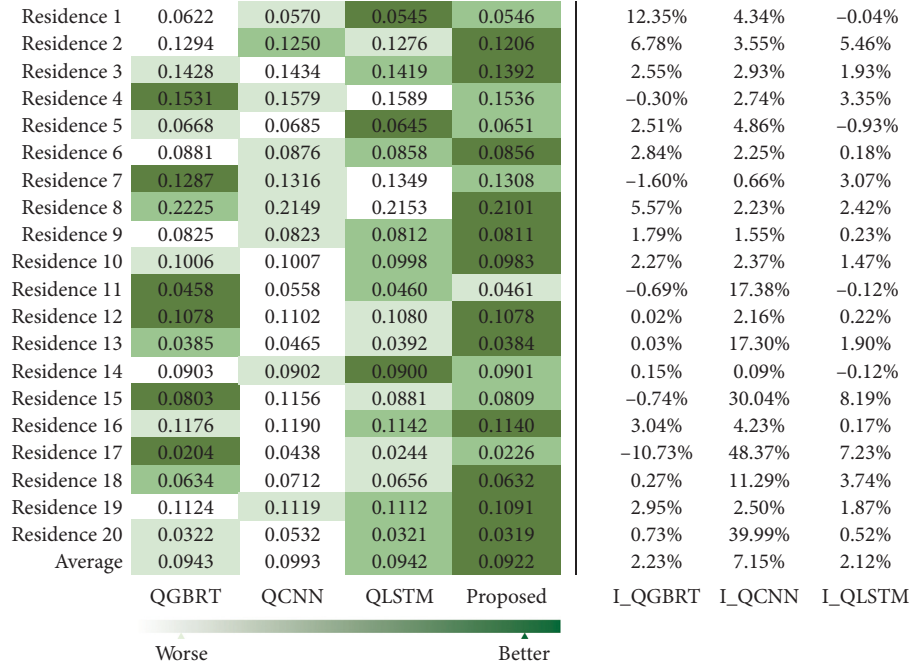


FIGURE 8: Heat map with AQS metrics for evaluation of four models. QGBRT, QCNN, QLSTM, and our proposed model in X-axis correspond to AQS for different models, while I_QGBRT, I_QCNN, and I_QLSTM represent the AQS improvement ratio compared with the proposed model. Y-axis denotes different residences. Results in each residence are described in the same row, where darker green shades imply a better performance.

TABLE 2: Average quantile score from 0.1 to 0.9.

| Quantiles | QGBRT | QCNN | QLSTM | Proposed | I_QGBRT (%) | I_QCNN (%) | I_QLSTM (%) |
|-----------|--------|--------|--------|----------|-------------|------------|-------------|
| 0.1 | 0.036 | 0.0393 | 0.0365 | 0.0359 | 0.28 | 8.65 | 1.64 |
| 0.2 | 0.0634 | 0.068 | 0.0643 | 0.0634 | 0.00 | 6.76 | 1.40 |
| 0.3 | 0.0856 | 0.0909 | 0.0865 | 0.0853 | 0.35 | 6.16 | 1.39 |
| 0.4 | 0.1028 | 0.1087 | 0.1032 | 0.1022 | 0.58 | 5.98 | 0.97 |
| 0.5 | 0.1151 | 0.121 | 0.115 | 0.1138 | 1.13 | 5.95 | 1.04 |
| 0.6 | 0.1218 | 0.128 | 0.1213 | 0.1197 | 1.72 | 6.48 | 1.32 |
| 0.7 | 0.1223 | 0.1282 | 0.1214 | 0.1189 | 2.78 | 7.25 | 2.06 |
| 0.8 | 0.1136 | 0.1186 | 0.1124 | 0.1085 | 4.49 | 8.52 | 3.47 |
| 0.9 | 0.0877 | 0.0911 | 0.0869 | 0.0818 | 6.73 | 10.21 | 5.87 |
| Average | 0.0943 | 0.0993 | 0.0942 | 0.0922 | 2.23 | 7.15 | 2.12 |

results of QGBRT and QLSTM are acceptable, while QCNN runs unsatisfactory. Taking AACE of 20 residences into account, the proposed method has an average growth of 34.71%, 75.22%, and 32.44% compared with QGBRT, QCNN, and QLSTM giving proof of the reliability in our proposed model significantly.

Figure 10 depicts the heat map of 80% interval PINAW. The QCNN acquires the best sharpness with 12 cases in 20 residences, and the performance of PINAW is also compelling. PINAW provides measurement on how compact PI is. If AACEs of different models are close, lower PI given by PINAW will be more instructive. QCNN's AACE is relatively poor than our proposed model, but PINAW serves more better, indicating that QCNN improves sharpness by sacrificing reliability. This result is not desirable unless the forecasting target explicitly requires more compact PIs. When PINAW and AACE come into conflicts, additional

comprehensive indicators, such as AQS, could be selected for accurate evaluation.

QGBRT and our proposed model obtain more sophisticated achievements rather than QLSTM. Most results of QLSTM are quite stable and similar with QGBRT, but the performance of our proposed model gives more fluctuations. In perspective of average improvement ratio, PINAW of our method in improved by 1.51%, 7.53%, and 1.87% relative to QGBRT, QCNN, and QLSTM, respectively. Considering the significant improvement in reliability, the deterioration of our proposed model in sharpness is still within an acceptable limit.

Figure 11 gives the coverage probability of the interval between each estimated quantile, and each interval corresponds to a bar. 10 intervals are generated from 9 quantiles (from 0-0.1 to 0.9-1), and the actual load values in the testing set have a uniform probability falling into these 10 intervals.

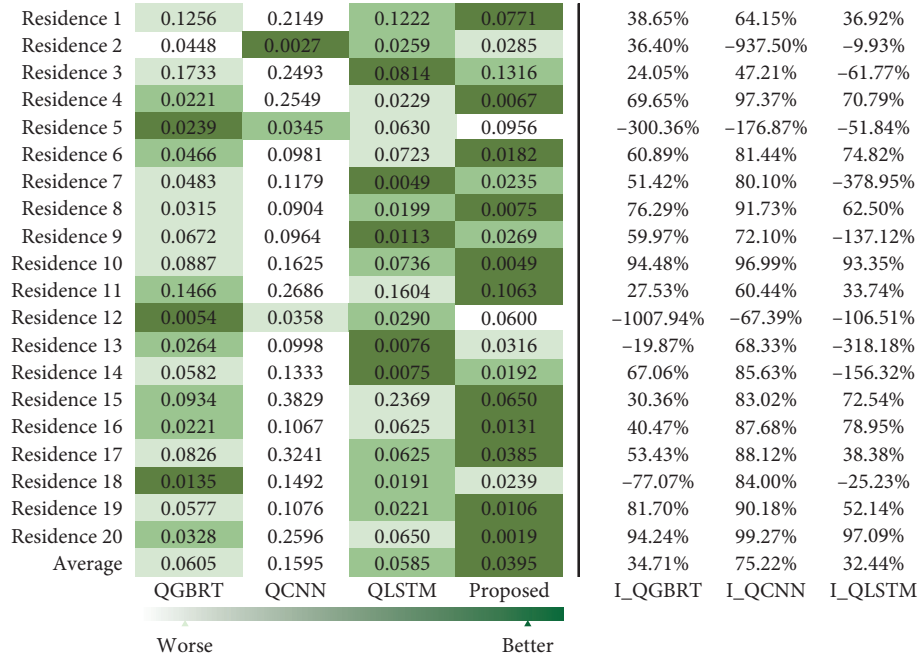


FIGURE 9: Heat map of 80% interval AACE for four models. QGBRT, QCNN, QLSTM, and our proposed model in X-axis correspond to AACE for different models, while I_QGBRT, I_QCNN, and I_QLSTM represent the AACE improvement ratio of our proposed model over other three models. Y-axis expresses different 20 residences. AACEs of the same residence are listed in the same row, where darker green shades imply a better performance.

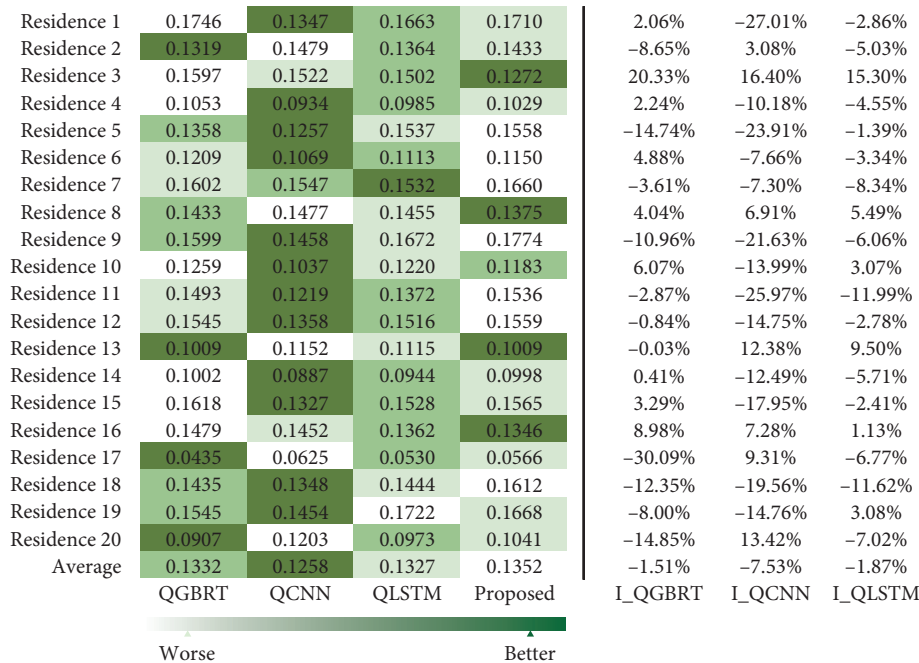


FIGURE 10: Heat map of the 80% PINAW for four models. QGBRT, QCNN, QLSTM, and our proposed model in X-axis correspond to PINAW, respectively, while I_QGBRT, I_QCNN, and I_QLSTM denote PINAW improvement ratio compared with our method over other models. Y-axis represents different residences. Results of each residence are compared in the same row, where darker green shades imply a better performance.

We normalized this probability according to the quantile width of intervals (i.e., 0.1); hence, each bar should be close to 1. The proposed model in Figure 11(d) provides the

minimum along Y-axis demonstrating that the quantile forecasting of our proposed model is reasonably interpreted in multiple residential data. Specifically, QGBRT, QCNN,

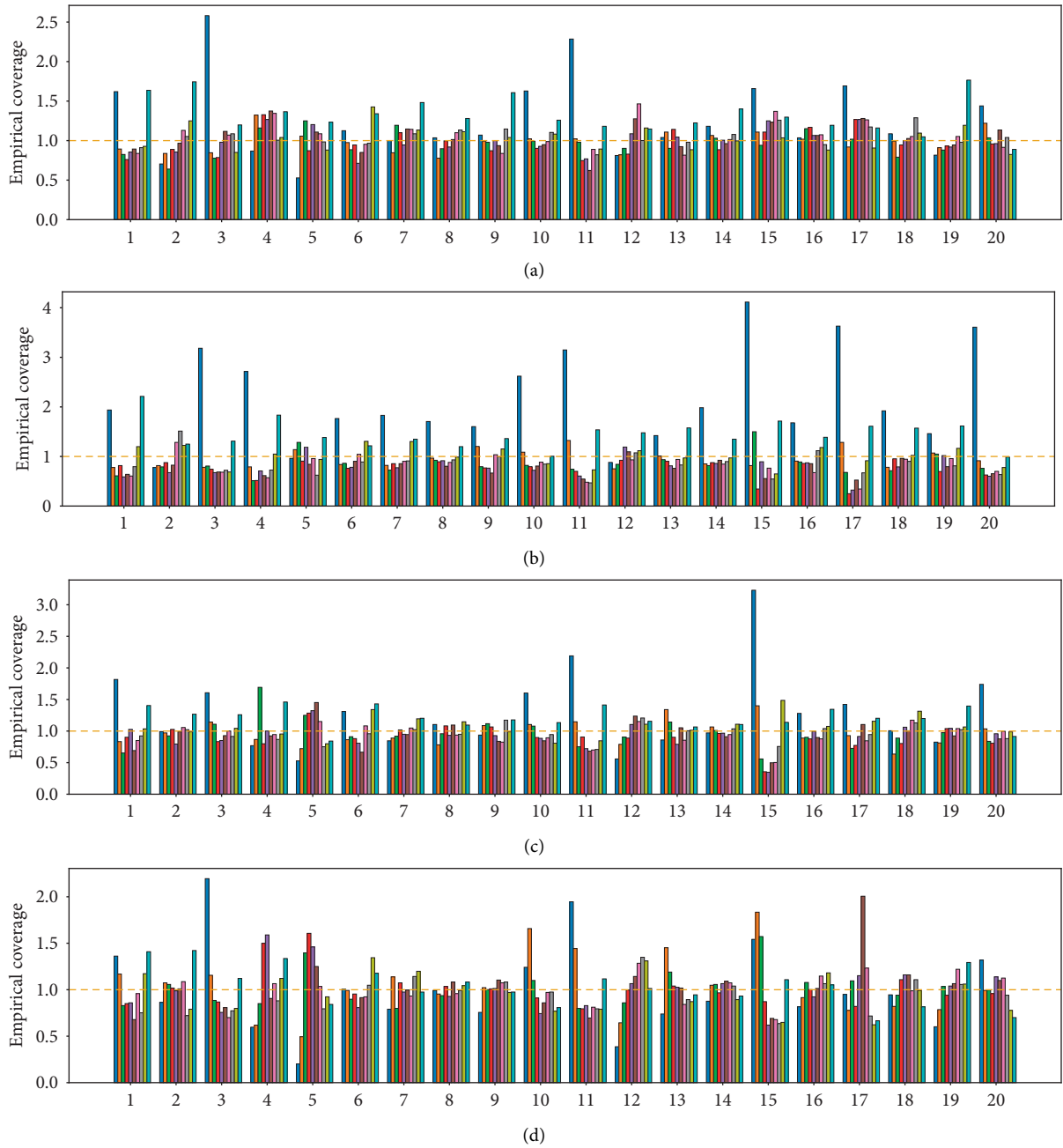


FIGURE 11: The empirical coverage of (a) QGBRT, (b) QCNN, (c) QLSTM, and (d) our proposed model. Y-axis denotes normalized empirical coverage and X-axis represents different residences with 10 bars individually corresponding to 10 intervals. A bar with height close to 1 indicates a better empirical coverage.

QLSTM, and our proposed model handle 2, 7, 2, and 1 bars, respectively. In general, most bars in QGBRT, QLSTM, and the proposed model tend to be close to 1.

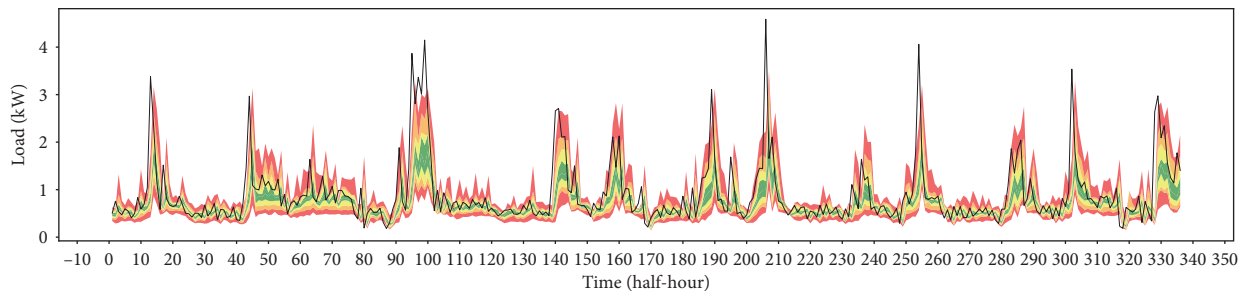
A strange phenomenon appeared that bars' average height of residence 1 and residence 15 in Figure 11(a) is larger than 1. For a special example, the forecasting results of quantiles 0.1, 0.2, and 0.3 are 1 kW, 0.8 kW, and 1.2 kW and the ground truth is 0.9 kW. The actual value is located in the interval 0-0.1 and 0.1-0.2 at the same time, which belongs to an inversion error of inverted quantiles. Table 3 reflects the number of errors in 20 residential quantile forecasting, where the number

of errors in our model is remarkably lower except residence 4 and 13. It demonstrates the superiority of our model in perspective of the regularity for quantiles' arrangement. In particular, QGBRT performed well in AQS, AACE, and PINAW but generated great errors, which could be influenced by the separated computing mechanism of quantile forecasting. Inversion errors become a wonderful complement to classic metrics of reliability, sharpness, and resolution, which avoid confusing results in the evaluation process.

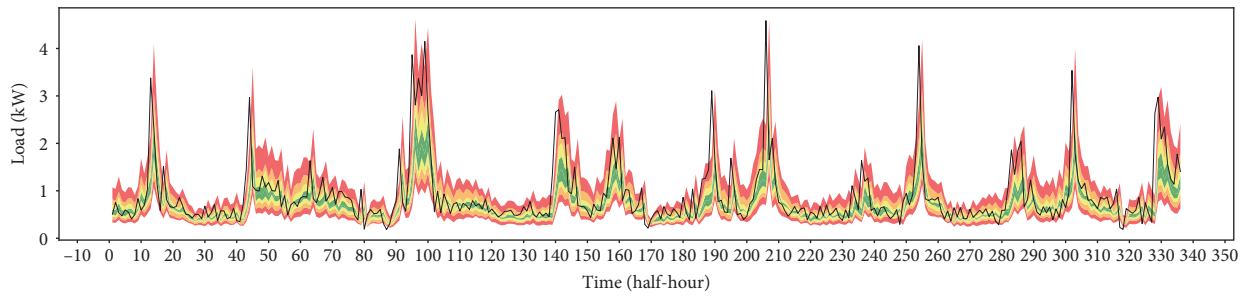
Figure 12 plots the actual load and the quantile forecasting of residence 2 in one week. Eight intervals among 0.1

TABLE 3: Inversion error statistics.

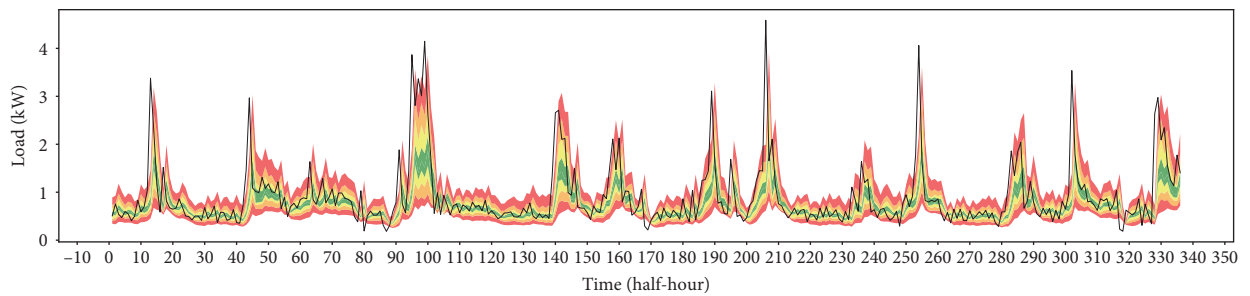
| Residence | QGBRT | QCNN | QLSTM | Proposed |
|-----------|-------|-------|-------|----------|
| 1 | 1854 | 1791 | 1989 | 407 |
| 2 | 476 | 478 | 385 | 311 |
| 3 | 6825 | 3623 | 4541 | 633 |
| 4 | 5256 | 918 | 2913 | 1700 |
| 5 | 1187 | 1465 | 570 | 87 |
| 6 | 1559 | 3222 | 2926 | 619 |
| 7 | 4223 | 2077 | 758 | 189 |
| 8 | 1215 | 1253 | 525 | 142 |
| 9 | 2906 | 1771 | 703 | 125 |
| 10 | 3913 | 2990 | 1225 | 440 |
| 11 | 1726 | 3309 | 754 | 271 |
| 12 | 2435 | 1388 | 836 | 438 |
| 13 | 812 | 1003 | 209 | 221 |
| 14 | 3377 | 2041 | 738 | 380 |
| 15 | 7411 | 3160 | 6991 | 857 |
| 16 | 2930 | 3172 | 1879 | 1050 |
| 17 | 7634 | 2486 | 286 | 269 |
| 18 | 2165 | 3475 | 1947 | 449 |
| 19 | 2066 | 3003 | 1107 | 561 |
| 20 | 1957 | 2227 | 475 | 398 |
| Total | 61927 | 44852 | 31757 | 9547 |



(a)



(b)



(c)

FIGURE 12: Continued.

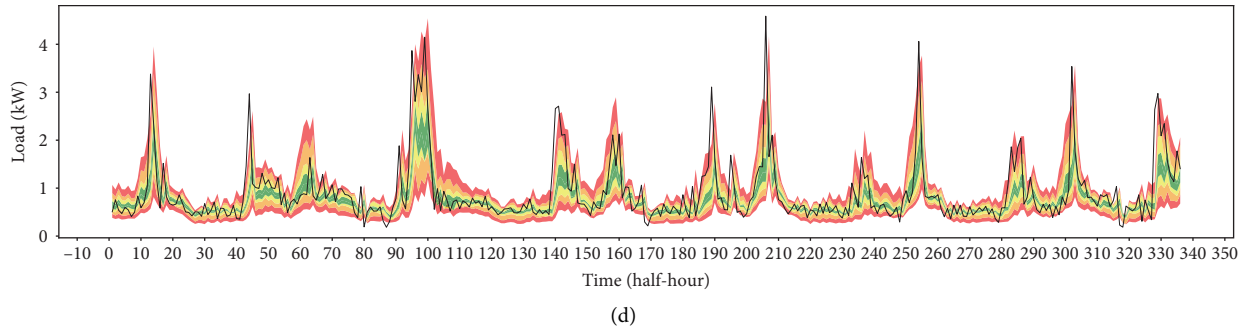


FIGURE 12: Forecasting results and ground truth of residence 2 for one week. X-axis expresses time with a half-hour unit, and Y-axis represents electricity load. Eight intervals between 0.1 and 0.9 are shown in different colors, where the solid black lines represent the ground truth. (a) QGBRT, (b) QCNN, (c) QLSTM, and (d) our proposed model.

TABLE 4: AQS comparison of three time-coding methods.

| Residence | Nature coding | One-hot coding | Periodic coding | I_nature (%) | I_one-hot (%) |
|-----------|---------------|----------------|-----------------|--------------|---------------|
| 1 | 0.0604 | 0.0592 | 0.0546 | 9.67 | 7.83 |
| 2 | 0.1261 | 0.1226 | 0.1206 | 4.40 | 1.67 |
| 3 | 0.1401 | 0.1449 | 0.1392 | 0.70 | 3.94 |
| 4 | 0.1823 | 0.1585 | 0.1536 | 15.75 | 3.10 |
| 5 | 0.0778 | 0.0666 | 0.0651 | 16.25 | 2.15 |
| 6 | 0.0885 | 0.0951 | 0.0856 | 3.29 | 9.99 |
| 7 | 0.1413 | 0.1378 | 0.1308 | 7.49 | 5.12 |
| 8 | 0.2379 | 0.2114 | 0.2101 | 11.66 | 0.60 |
| 9 | 0.0888 | 0.0895 | 0.0811 | 8.68 | 9.42 |
| 10 | 0.1091 | 0.1018 | 0.0983 | 9.91 | 3.47 |
| 11 | 0.0523 | 0.0475 | 0.0461 | 11.89 | 2.91 |
| 12 | 0.1176 | 0.1116 | 0.1078 | 8.34 | 3.38 |
| 13 | 0.0433 | 0.0401 | 0.0384 | 11.27 | 4.21 |
| 14 | 0.0983 | 0.0946 | 0.0901 | 8.26 | 4.67 |
| 15 | 0.0933 | 0.0896 | 0.0809 | 13.27 | 9.75 |
| 16 | 0.1188 | 0.1239 | 0.1140 | 4.03 | 8.00 |
| 17 | 0.0259 | 0.0253 | 0.0226 | 12.89 | 10.63 |
| 18 | 0.0707 | 0.0630 | 0.0632 | 10.63 | -0.30 |
| 19 | 0.1200 | 0.1204 | 0.1091 | 9.09 | 9.34 |
| 20 | 0.0351 | 0.0349 | 0.0319 | 9.01 | 8.51 |
| Average | 0.1014 | 0.0969 | 0.0922 | 9.10 | 4.91 |

and 0.9 quantiles are shown in different colors, while solid black lines represent the ground truth. It is found that algorithms based on historical load try to explore the short-term trend of the load as far as possible. As a result of the volatility of the profile, there is always a delay when the peak of the load is being captured. Besides, a peak load corresponding to the 90th–100th half-hour should be paid more attentions and 80% PI of all four models failed to capture the first peak. Especially, our proposed model significantly learned discriminative features in sequences, but QGBRT and QLSTM were not competent. In addition, before several peaks in the 60th, 210th, 255th, and 305th half-hour, our proposed model improved the result of prediction dramatically, and consequently its 80% PI remarkably caught load trend. However, other models could not attain this achievement, manifesting superiority of our model in dealing with the peak point. This capability should be attributed to the periodic coding that provides our model with advanced periodical characteristics at certain moments.

5.3. Evaluation of Time-Coding Method. In order to evaluate the effectiveness of periodic coding, we compared AQS of periodic coding, natural coding, and one-hot coding on the proposed network architecture, as shown in Table 4. I_nature and I_one-hot denote the improvement ratio of periodic coding over natural coding and one-hot coding. The AQS of natural coding grew quickly, even higher than QCNN, indicating that it is not competent to improve AQS. The AQS of one-hot coding served preferred performance rather than natural coding, yet the degree is still limited. Specifically, the average AQS of periodic coding achieved the best results, 9.10% and 4.91%, respectively, lower than other two approaches remarkably.

5.4. Efficiency. Computation efficiency and memory cost of compared models are shown in Table 5. Deep neural networks serve numerous parameters and produce massive computation costs. Fortunately, we can address this issue by

TABLE 5: Average efficiency comparison of 20 residences.

| Item | QGBRT | QCNN | QLSTM | Proposed | R_QLSTM |
|-----------------------------|-------|---------|---------|----------|---------|
| Training time per epoch (s) | None | 24 | 209 | 26 | 87.56% |
| Testing time (s) | 0.18 | 0.88 | 16.27 | 0.89 | 94.53% |
| Training convergence epochs | None | 77 | 68 | 66 | — |
| Total training time (s) | 1,517 | 1,814 | 14,212 | 1,716 | 87.93% |
| Parameters | None | 184,681 | 330,889 | 187,769 | — |

adopting professional graphic cards; thus, the deep neural network models in our studies were run on the NVIDIA P4 cards by default. All comparison experiments are qualified with the same configuration.

In Table 5, each training epoch of QLSTM provides larger computation cost than QCNN and our proposed model, resulting from higher dependency in steps where the entire iterative process is impossible to be deployed in parallel approach. Similarly, when run on 2328 samples for testing, QLSTM nearly gave 18 times costs over the CNN-based model. Especially, column R_QLSTM expresses that our proposed model reduces the cost of training, testing, and total training cost by 87.56%, 94.53%, and 87.93%, respectively, over the QLSTM in each epoch. Efficiency of QGBRT is relatively preferable than others owing to its simple mechanism of extracting features, which only took 0.18 s. Both QLSTM and our proposed model adopt the S2S structure that enhances the flow of gradients and the process of convergence. QGBRT, QCNN, and our proposed model were close in the cost of training, and QLSTM took much longer. It is easily found that the cost gap of QGBRT between the training and testing process is larger than others due to QGBRT’s estimators without the parallel mechanism. Unlike neural networks, each quantile of the QGBRT requires separate training operation that leads to an increase in the training costs of the QGBRT. Besides, on the scale of parameters, the achievements obtained by the proposed model require less parameters than QLSTM, providing notable efficiency in comparison.

6. Conclusion

Residential load forecasting is important for many entities in the electricity market, but the load profile of individual residence shows great volatilities and uncertainties. Due to the difficulty in producing reliable point forecasts, probabilistic load forecasting becomes a research focus that could explore the volatility and uncertainty by intervals, density, or quantiles. In this paper, we propose a unified quantile regression deep neural network with time-cognition for tackling this challenging issue. At first, a deep convolutional neural network with multiscale dilated convolutions is proposed for extracting more significant features from the historical load sequence. In addition, a periodical coding method is devised to mark the input sequence for capturing regular load pattern. Then, features generated from both of subnetworks are fused and fed into the forecasting model with an end-to-end manner. At last, forecasts of multiple quantiles are directly outputted in one shot.

To ensure the accuracy of the experiments, we conducted experiments on 20 randomly selected residential data for

evaluations. Sufficient experiments compared our proposed model with several state-of-the-art works obtaining comprehensive conclusions. Metrics such as AQS, AACE, and PINAW are used to quantitatively evaluate subjects in perspectives of reliability and sharpness. In addition, we also paid more attentions on empirical coverage and quantile inversion error to provide additional measures on performances. Experimental results showed that our proposed model achieves the best results in the AQS, AACE, and inversion errors, and especially the average AACE of our model is increased by 34.71%, 75.22%, and 32.44%, respectively, compared with QGBRT, QCNN, and QLSTM, indicating that the proposed network has remarkably excellent reliability. In addition, we analyzed the efficiency of subjects in computation and found that our proposed model has lower burden training and testing cost, which reflects faster time response rather than QLSTM arguing that our model serves promising prospects in practical applications.

The CNN-based deep learning models have achieved many state-of-the-art results in sequential problems recently. Through the experimental results of our work, the well-designed CNN model can not only achieve high precision but also approximate the traditional machine learning algorithm in efficiency, serving a good practical application prospect. Technologies such as transformable convolution and attention mechanism have enormous potentials in load sequence forecasting, and we will continue to explore this field in future work.

Data Availability

The CBT data (open dataset) used to support the findings of this study could be accessed in the ISSDA website: <http://www.ucd.ie/issda/data/commissionforenergyregulationcer/>.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This research was funded by the State Grid Corporation of China (2018YF-51 and 2019YF-40).

References

- [1] T. Hong and S. Fan, “Probabilistic electric load forecasting: a tutorial review,” *International Journal of Forecasting*, vol. 32, no. 3, pp. 914–938, 2016.

- [2] Ö. F. Ertugrul, "Forecasting electricity load by a novel recurrent extreme learning machines approach," *International Journal of Electrical Power & Energy Systems*, vol. 78, pp. 429–435, 2016.
- [3] C. Keerthisinghe, V. Gregor, and A. C. Chapman, "A fast technique for smart home management: Adp with temporal difference learning," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3291–3303, 2016.
- [4] A. Pratt, D. Krishnamurthy, M. Ruth, H. Wu, M. Lunacek, and P. Vaynshenk, "Transactive home energy management systems: the impact of their proliferation on the electric grid," *IEEE Electrification Magazine*, vol. 4, no. 4, pp. 8–14, 2016.
- [5] T. Morstyn, N. Farrell, S. J. Darby, and M. D. McCulloch, "Using peer-to-peer energy-trading platforms to incentivize prosumers to form federated power plants," *Nature Energy*, vol. 3, no. 2, pp. 94–101, 2018.
- [6] Yi Wang, Q. Chen, H. Tao, and C. Kang, "Review of smart meter data analytics: applications, methodologies, and challenges," *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 3125–3148, 2019.
- [7] Y. Wang, Q. Chen, M. Sun, C. Kang, and Q. Xia, "An ensemble forecasting method for the aggregated load with subprofiles," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3906–3908, 2018.
- [8] B. Stephen, X. Tang, P. R. Harvey, S. Galloway, and K. I. Jennett, "Incorporating practice theory in sub-profile models for short term aggregated residential load forecasting," *IEEE Transactions on Smart Grid*, vol. 8, no. 4, pp. 1591–1598, 2017.
- [9] S. Humeau, T. K. Wijaya, M. Vasirani, and K. Aberer, "Electricity load forecasting for residential customers: exploiting aggregation and correlation between households," in *Proceedings of the 2013 Sustainable Internet and ICT for Sustainability (SustainIT)*, pp. 1–6, Palermo, Italy, October 2013.
- [10] R. E. Edwards, J. New, and L. E. Parker, "Predicting future hourly residential electrical consumption: a machine learning case study," *Energy and Buildings*, vol. 49, pp. 591–603, 2012.
- [11] R. Ahmadiyahangar, H. Tobias, A. Rosin, T. Korötko, and J. Martins, "Residential load forecasting for flexibility prediction using machine learning-based regression model," in *Proceedings of the 2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*, pp. 1–4, Genova, Italy, June 2019.
- [12] A. Tascikaraoglu and B. M. Sanandaji, "Short-term residential electric load forecasting: a compressive spatio-temporal approach," *Energy and Buildings*, vol. 111, pp. 380–392, 2016.
- [13] C.-N. Yu, P. Mirowski, and T. K. Ho, "A sparse coding approach to household electricity demand forecasting in smart grids," *IEEE Transactions on Smart Grid*, vol. 8, no. 2, pp. 738–748, 2017.
- [14] Li Pan, B. Zhang, W. Yang, and R. Rajagopal, "A sparse linear model and significance test for individual consumption prediction," *IEEE Transactions on Power Systems*, vol. 32, no. 6, pp. 4489–4500, 2017.
- [15] A. Almalaq and G. Edwards, "A review of deep learning methods applied on load forecasting," in *Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 511–516, Cancun, Mexico, December 2017.
- [16] F. Abbas, D. Feng, S. Habib, U. Rahman, A. Rasool, and Z. Yan, "Short term residential load forecasting: an improved optimal nonlinear auto regressive (narx) method with exponential weight decay function," *Electronics*, vol. 7, no. 12, p. 432, 2018.
- [17] H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting—a novel pooling deep RNN," *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 5271–5280, 2018.
- [18] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on lstm recurrent neural network," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 841–851, 2019.
- [19] T. Hossen, A. S. Nair, R. A. Chinnathambi, and P. Ranganathan, "Residential load forecasting using deep neural networks (DNN)," in *Proceedings of the 2018 North American Power Symposium (NAPS)*, pp. 1–5, Fargo, ND, USA, September 2018.
- [20] H. Tao, J. Wilson, and J. Xie, "Long term probabilistic load forecasting and normalization with hourly information," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 456–462, 2014.
- [21] R. J. Hyndman and S. Fan, "Density forecasting for long-term peak electricity demand," *IEEE Transactions on Power Systems*, vol. 25, no. 2, pp. 1142–1153, 2010.
- [22] P. E. McSharry, S. Bouwman, and G. Bloemhof, "Probabilistic forecasts of the magnitude and timing of peak electricity demand," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 1166–1172, 2005.
- [23] J. Xie and H. Tao, "Temperature scenario generation for probabilistic load forecasting," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 1680–1687, 2016.
- [24] S. Arora and J. W. Taylor, "Forecasting electricity smart meter data using conditional kernel density estimation," *Omega*, vol. 59, pp. 47–59, 2016.
- [25] A. Mousa, M. Mohammadi, M. Rastegar, and K. Amin, "Probabilistic deep neural network price forecasting based on residential load and wind speed predictions," *IET Renewable Power Generation*, vol. 13, no. 11, pp. 1840–1848, 2019.
- [26] Y. Yang, S. Li, W. Li, and M. Qu, "Power load probability density forecasting using Gaussian process quantile regression," *Applied Energy*, vol. 213, pp. 499–509, 2018.
- [27] P. Gaillard, Y. Goude, and R. Nedellec, "Additive models and robust aggregation for gefcom2014 probabilistic electric load and electricity price forecasting," *International Journal of Forecasting*, vol. 32, no. 3, pp. 1038–1050, 2016.
- [28] J. Xie, H. Tao, T. Laing, and C. Kang, "On normality assumption in residual simulation for probabilistic load forecasting," *IEEE Transactions on Smart Grid*, vol. 8, no. 3, pp. 1046–1053, 2017.
- [29] B. Liu, J. Nowotarski, H. Tao, and R. Weron, "Probabilistic load forecasting via quantile regression averaging on sister forecasts," *IEEE Transactions on Smart Grid*, vol. 8, no. 2, pp. 730–737, 2015.
- [30] M. Shepero, D. Meer, J. Munkhammar, and J. Widén, "Residential probabilistic load forecasting: a method using Gaussian process designed for electric load data," *Applied Energy*, vol. 218, pp. 159–172, 2018.
- [31] A. Gerossier, R. Girard, G. Kariniotakis, and A. Michiorri, "Probabilistic day-ahead forecasting of household electricity demand," *CIRED—Open Access Proceedings Journal*, vol. 2017, no. 1, pp. 2500–2504, 2017.
- [32] S. Ben Taieb, R. Huser, R. J. Hyndman, and M. G. Genton, "Forecasting uncertainty in electricity smart meter data by boosting additive quantile regression," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2448–2455, 2016.
- [33] W.-C. Hong, "Electric load forecasting by seasonal recurrent svr (support vector regression) with chaotic artificial bee colony algorithm," *Energy*, vol. 36, no. 9, pp. 5568–5578, 2011.

- [34] W. Li, X. Yang, H. Li, and L. Su, "Hybrid forecasting approach based on grnn neural network and svr machine for electricity demand forecasting," *Energies*, vol. 10, no. 1, p. 44, 2017.
- [35] Yi Wang, N. Zhang, Q. Chen, D. S. Kirschen, Li Pan, and Q. Xia, "Data-driven probabilistic net load forecasting with high penetration of behind-the-meter PV," *IEEE Transactions on Power Systems*, vol. 33, no. 3, pp. 3255–3264, 2018.
- [36] S. Papadopoulos and I. Karakatsanis, "Short-term electricity load forecasting using time series and ensemble learning methods," in *Proceedings of the 2015 IEEE Power and Energy Conference at Illinois (PECI)*, pp. 1–6, IEEE, Champaign, IL, USA, February 2015.
- [37] G. Juberias, J. Garcia Moreno, and C. Mendivil, "A new ARIMA model for hourly load forecasting," in *Proceedings of the 1999 IEEE Transmission and Distribution Conference (Cat. No. 99CH36333)*, vol. 1, pp. 314–319, IEEE, New Orleans, LA, USA, April 1999.
- [38] W. Li and Z. G. Zhang, "Based on time sequence of arima model in the application of short-term electricity load forecasting," in *Proceedings of the 2009 International Conference on Research Challenges in Computer Science*, pp. 11–14, IEEE, Shanghai, China, December 2009.
- [39] K. Greff, R. K. Srivastava, K. Jan, B. R. Steunebrink, and J. Schmidhuber, "LSTM: a search space odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2017.
- [40] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," 2015, <https://arxiv.org/abs/1506.00019>.
- [41] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu, and C. Kang, "Probabilistic individual load forecasting using pinball loss guided lstm," *Applied Energy*, vol. 235, pp. 10–20, 2019.
- [42] A. van den Oord, D. Sander, H. Zen et al., "Wavenet: a generative model for raw audio," 2016, <https://arxiv.org/abs/1609.03499>.
- [43] N. Kalchbrenner, L. Espeholt, K. Simonyan, A. van den Oord, A. Graves, and K. Kavukcuoglu, "Neural machine translation in linear time," 2016, <https://arxiv.org/abs/1610.10099>.
- [44] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, pp. 933–941, Sydney, Australia, August 2017.
- [45] M. Cai, M. Pipattanasomporn, and S. Rahman, "Day-ahead building-level load forecasts using deep learning vs. traditional time-series techniques," *Applied Energy*, vol. 236, pp. 1078–1088, 2019.
- [46] X. Dong, L. Qian, and L. Huang, "Short-term load forecasting in smart grid: a combined cnn and k-means clustering approach," in *Proceedings of the 2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 119–125, IEEE, Jeju, South Korea, February 2017.
- [47] Z. Deng, B. Wang, Y. Xu, T. Xu, C. Liu, and Z. Zhu, "Multi-scale convolutional neural network with time-cognition for multi-step short-term load forecasting," *IEEE Access*, vol. 7, pp. 88058–88071, 2019.
- [48] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 156–165, Honolulu, HI, USA, July 2017.
- [49] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, <https://arxiv.org/abs/1803.01271>.
- [50] Z. Yu, Z. Niu, W. Tang, and Q. Wu, "Deep learning for daily peak load forecasting—a novel gated recurrent neural network combining dynamic time warping," *IEEE Access*, vol. 7, pp. 17184–17194, 2019.
- [51] Y. R. Yue and H. Rue, "Bayesian inference for additive mixed quantile regression models," *Computational Statistics & Data Analysis*, vol. 55, no. 1, pp. 84–96, 2011.
- [52] M. Abadi, B. Paul, J. Chen et al., "Tensorflow: a system for large-scale machine learning," in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pp. 265–283, San Diego, CA, USA, November 2016.
- [53] M. Rafiei, T. Niknam, J. Aghaei, M. Shafie-Khah, and J. P. S. Catalao, "Probabilistic load forecasting using an improved wavelet neural network trained by generalized extreme learning machine," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6961–6971, 2018.
- [54] H. Quan, D. Srinivasan, and A. Khosravi, "Short-term load and wind power forecasting using neural network-based prediction intervals," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 303–315, 2013.




Hindawi

Submit your manuscripts at
www.hindawi.com

